

Diplomarbeit

von Cand. Dipl. Ing. Kartografie (FH) Heiko Mundle

Methana 3D

**Eine kartografische Online-3D-Plattform mit
Datenbankanbindung der Halbinsel Methana (Griechenland)**

Prof. Dr. Ch. Herrmann, FH Karlsruhe, Prof. Dr. L. Hurni ETH Zürich

FH-Karlsruhe – Hochschule für Technik
Studiengang
FB Geoinformationswesen
Studiengang Kartographie und Geomatik

FH Karlsruhe-Hochschule für Technik
FB Geoinformationswesen
Studiengang Kartographie und Geomatik
Prof. Dr. Ch. Herrmann

Diplomarbeit

für Heiko Mundle

Thema: Methana 3D – ein kartographische Online-3D-Plattform mit Datenbankanbindung der Halbinsel Methana (Griechenland)

Einführung:

Das Kommunikationsmedium Internet gewinnt für die Kartographie immer mehr an Bedeutung, da es eine Vielzahl von Möglichkeiten zur Verarbeitung, Visualisierung und Distribution raumbezogener Informationen eröffnet. Die Möglichkeiten zur Bereitstellung von dreidimensionalen Geländemodellen im Netz sind bisher nur in Ansätzen verwirklicht. Seit kurzer Zeit erlauben moderne Beschreibungssprachen wie z. B. VRML oder X3D eine kompakte Online-Übertragung dieser Modelle.

Für die Kartographie ist dieser Ansatz besonders Interessant, weil der Nutzer auf neuartige, unmittelbar verständliche Visualisierungen zugreifen kann. Gegenüber der Offline CD-ROM ist die Online-Plattform ein ständig aktualisierbares, weltweit verfügbares Medium.

Aufgabe:

Am IKA der ETH Zürich soll bei Prof. Dr. L. Hurni eine kartographische Online-3D-Plattform mit Datenbankanbindung der Halbinsel Methana (GR) bearbeitet werden.

Dazu ist die für Visualisierung von 3D Objekten speziell entwickelte Beschreibungssprache Virtual Reality Modeling Language (VRML) oder X3D zu benützen. Sie wurden betriebsystemneutral konzipiert und zeichnen sich durch geringe Übertragungsraten aus. Der Nutzer kann die Objekte in einem Browser mit einem Plug-in aus beliebigen Richtungen betrachten.

Dem Nutzer soll es möglich werden, via Internet schon vor dem Reiseantritt einen virtuellen Eindruck von der Halbinsel Methana zu erhalten. Für eine erweiterte Interaktivität ist die Beziehung Client – Datenbankserver via Internet zu benützen.

Die Aufgaben der Diplomarbeit gliedert sich somit in folgende Teilbereiche:

1. Konzeption einer graphischen, benutzerfreundlichen Oberfläche,
2. parallele Steuerung der 3D-Plattform mit dem External Authoring Interface (EAI) aus HTML,
3. der Entwicklung des digitalen Höhenmodells (DHM) von Methana,
4. Generierung der Einzelobjekte ab Datenbank und ihre Verlinkung mit Zusatzinformationen (auch externe Links zu vorhandenen Webseiten),
5. Verschneidung des DHM mit verschiedenen Oberflächenbedeckungselementen,
6. Realisierung eines Prototyps, der auf weitere Regionen übertragen werden kann.

Endprodukt:

Beschreibender Text, Prototyp inkl. dokumentiertem Sourcecode, Daten auf geeignetem Datenträger (CD-ROM), die sowohl auf Windows und Mac OS Browser lesbar sind.

Dauer der Arbeit: 6 Monate

Ausgabe der Arbeit:

Abgabe der Arbeit:

Erklärung

Hiermit versichere ich, dass ich diese Diplomarbeit selbständig verfasst und keine andern als die von mir angegebenen Quellen und Hilfsmittel benutzt habe.

Karlsruhe, den 13. 11. 1999

Vorwort

Diese Diplomarbeit entstand im Zeitraum von Juni bis November 1999 am Institut für Kartographie (IKA) an der ETH Zürich unter der Leitung von Prof. Dr. Lorenz Hurni und Prof. Dr. Herrmann von der FH Karlsruhe, Fachbereich Geoinformationswesen.

Ich danke, dass mir am IKA ein Arbeitsplatz zur Verfügung gestellt wurde und ich die vorhandenen Daten von Methana ohne Einschränkung weiterverwenden konnte. Lorenz Hurni sage ich Danke für seine Begeisterung für das Projekt und dass er mich mit seinen Ortskenntnissen unterstützt hat.

Ganz besonderer Dank gilt Tobias Schorr, der mich ebenfalls mit Insiderwissen versorgt hat und der mir seine Texte und hervorragenden Fotografien zur Verfügung gestellt hat. Vielleicht können wir die Projekte methana.de und Methana 3D noch zu einem Ganzen vereinen.

Danken will ich meinem Betreuer Andreas Neumann für seine Impulse, seine technischen Ratschläge und dass er so geduldig die Softwareadministration

gemacht hat. Prof. Herrmann danke ich, dass er an mich geglaubt hat und mich gefördert hat.

Ein dickes Dankeschön an Euch alle, die Ihr mich unterstützt habt:

Daniel Bielser, Cordula Bietzker fürs Korrekturlesen, Christoph Brandenberger für die Unterstützung bei der Dateikonvertierung, Sonja Gauch, Dorothea Graffe, Christian Häberling, Ernst Hutzler, Bernd Kamjunke fürs Korrekturlesen, Margrit Péquignot, Stefan Räber, Barbara Schneider für die Erste Hilfe bei Arc/Info, Volker Schniepp, Andrea Terribilini für die Anpassung Deiner Programme zum Berechnen von Höhenmodellen und die Tipps zur Objekterzeugung, Ulrike Werner, Daniela Winkler und und und ...

Grosser Dank geht an meine Familie, die mich in allen Belangen unterstützt hat.

Der grösste Dank gebührt meinem Schöpfer, der mich mit meinen Talenten und Fähigkeiten ausgestattet hat.

Karlsruhe, November 1999

Heiko Mundle

Inhaltsverzeichnis

	1. Kapitel	
Einleitung	1.1 Motivation	8
	1.2 Problemstellung	8
	1.3 Zielsetzung	8
	1.4 Aufbau der Arbeit	9
	2. Kapitel	
Definition	2.1 Kartografisch	10
	2.2 Online	10
	2.3 Plattform	10
	2.4 3D	10
	2.5 Datenbankanbindung	10
	2.6 Zusammenfassung	11
	3. Kapitel	
Werkzeuge	3.1 Werkzeuge zur Real-Time-3D-Visualisierung	12
	3.1.1 Eine kleine Markübersicht	12
	3.1.2 Werkzeuge zur Objektbetrachtung	12
	3.1.3 Werkzeuge für Single-User-Welten	13
	3.1.4 Werkzeuge für Multi-User-Welten	14
	3.1.5 Werkzeugfestlegung Real-Time-3D-Visualisierung	14
	3.2 Werkzeuge zur Datenbearbeitung	15
	4. Kapitel	
Einführung in VRML	4.1 Entstehung	17
	4.2 Aufbau	17
	4.3 Wichtige VRML-Charakteristika für Methana 3D	18
	4.4 Zukunft VRML	19
	5. Kapitel	
VRML in der Kartografie	5.1 Virtual Reality (VR)	20
	5.2 Der praktische Gebrauch	20
	5.3 GeoVRML Working Group	21
	5.4 GeoVRML 1.0	21
	5.5 Beispiele	22
	6. Kapitel	
Die Halbinsel Methana	6.1 Übersicht	24
	6.2 Eignung der Halbinsel Methana	24
	6.3 Quellen für Methana 3D	25
	7. Kapitel	
Objekte auf Methana 3D	7.1 Topografische Basiselemente	26
	7.1.1 Punkthafte Objekte	26
	7.1.2 Linienhafte Objekte	26
	7.1.3 Flächenhafte Objekte	26
	7.2 Thematische Elemente	26
	7.2.1 Geologie	26
	7.2.2 Archäologie	26
	7.2.3 Tourismus	26

	8. Kapitel	
Integration und Signaturierung der Objekte in Methana 3D	8.1 Das Grundgerüst, das DGM	27
	8.1.1 Grundlagen	27
	8.1.2 Umsetzung	25
	8.2 Farbe in der Welt, verschiedene Oberflächenfarben	28
	8.2.1 Grundlagen	28
	8.2.2 Umsetzung	28
	8.3 Linienhafte Objekte	29
	8.3.1 Grundlagen/Allgemeines	29
	8.3.2 Umsetzung	29
	8.4 Punkthafte Objekte	30
8.4.1 Allgemeines	30	
8.4.2 Umsetzung	30	
	9. Kapitel	
Datenbank für Methana 3D	9.1 Einführung	32
	9.2 Aufbau der DB	32
	9.3 Objekterzeugung aus der DB	33
	9.4 HTML-Erzeugung aus der DB	33
	9.5 HTML-Listen aus der DB	33
	10. Kapitel	
Funktionalität von Methana 3D	10.1 Übersicht	34
	10.2 VRML-Welt	34
	10.3 Hauptmenü	34
	10.4 Übersichtskarte	34
	10.5 Auswahllisten	36
	10.6 Inforahmen	36
	10.6.1 choose topic	36
	10.6.2 select by list	37
	10.6.3 help / refresh	37
	10.6.4 links	37
	10.6.5 about project	37
	10.7 Legende	37
	11. Kapitel	
Schlussanalyse	11.1 Problembetrachtung	38
	11.1.1 Probleme bei der Objekterzeugung	38
	11.1.2 Probleme mit klaren Lösungen	38
	11.1.3 Softwareprobleme	38
	11.1.4 Geschwindigkeitsprobleme	39
	11.2 Erweiterungsmöglichkeiten	39
	11.3 Übertragung	40
11.4 Zusammenfassung	40	
	Verzeichnisse	
	Abbildungsverzeichnis	41
	Literaturverzeichnis	43

	Anhang A	
Listing	1. AML	A 1
	2. Perl	A 13
	3. HTML	A 35
	4. Javascript	A 42
	5. SQL	A 49
	6. Perl – CGI	A 50
	7. VRML	A 55
	Anhang B	
Lt2VRML	1. Dokumentation	A 57
	2. Listing	A 58
	AML	A 58
	Perl	A 62

1. Einleitung

1.1 Motivation

Plastische Darstellungen der Erdoberfläche wurden in der Kartografie schon immer eingesetzt, um einem Interessierten eine bessere Vorstellung der räumlichen Gegebenheiten einer Landschaft zu vermitteln.

Schon vor Jahrhunderten wurden Reliefs geformt. Von Franz Ludwig Pfyffers Relief der Urschweiz schreibt A. BÜRGI 1998, S. 6, „dass gerade Reisende sich vom Pfyffer-Relief derart fasziniert zeigten, bot es doch einen Überblick über eine Landschaft, der so nicht einmal von einem noch so ideal gelegenen Aussichtspunkt im Gelände hätte genossen werden können“. Im Alpinen Museum in Bern kann man eine Vielzahl solcher Reliefs bewundern.

Zweidimensionale Zeichnungen können ebenfalls einen plastischen Eindruck vermitteln. Zu den bekanntesten zählen sicher die Bildstadtpläne von Friedrich BOLLMANN 1986 oder die „Panoramen“ von Heinrich C. BERANN 1886 (Abb. 1). Sie werden zur Orientierung in einer Stadt oder einem Wandergebiet, für Skipisten in den Alpen genutzt oder einfach als Poster an die Wand gehängt.

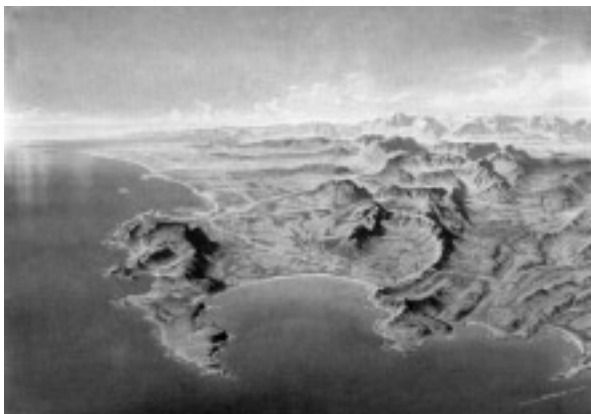


Abbildung 1: *Kapstadt mit Kap der Guten Hoffnung, False Bay und Stellenbosch. Gemalt von H. C. Berann.*

Heute findet man immer häufiger mit Computern erzeugte 3D-Darstellungen. Durch die rasante Hardwareentwicklung ist es möglich, diese Bilder in kürzester Zeit zu berechnen. Anwendungen, die uns in Echtzeit durch virtuelle Landschaften laufen lassen, werden möglich. Die Erweiterung der Darstellungen mit beliebiger Bewegung im Raum verhilft einem Betrachter, sich auf einem selbst gewählten Weg durch die Landschaft zu bewegen. Damit öffnen sich neue

Aspekte in der Computerkartografie, einem Interessierten ein Gebiet zu veranschaulichen. Ähnlich wie bei einem plastischen Relief kann er selbst steuern, wohin er sich bewegt und so sonst verdeckte Bereiche besuchen. Ausserdem wird noch weniger Abstraktionsvermögen benötigt, um eine Vorstellung über die räumlichen Beziehungen zu erhalten.

Die jüngste Entwicklung versucht, solche interaktiven 3D-Darstellungen mit Hilfe des Internets einem breiten Publikum zur Verfügung zu stellen.

1.2 Problemstellung

Anlass zur Diplomarbeit gaben verschiedene Probleme. Wie können geografische, dreidimensionale Informationen im Internet präsentiert werden? Auf welche Art kann ein durchschnittlicher PC diese Informationen in Real-Time darstellen? Welche Anforderungen werden an den Aufbau der Geometrie gestellt?

Wie kann im Internet Aufmerksamkeit erzeugt werden? Gibt es ungewöhnliche, nicht alltägliche Techniken?

Um Produktionskosten niedrig zu halten, ist es wichtig, dass rationelle Arbeitsmethoden angewandt werden. Wie kann es erreicht werden, dass die Ausgangsdaten mit möglichst wenig interaktiven Eingriffen verwendet werden können? Kann ein solcher Arbeitsablauf auch auf andere Projekte übertragen werden?

1.3 Zielsetzung

Ziel der Diplomarbeit ist der Prototyp eines Online-Informationssystem der Halbinsel Methana, mit dem einerseits bei potentiellen Urlaubern Interesse geweckt werden kann. Andererseits können die Reisenden schon vor Reiseantritt Methana einen virtuellen Besuch abstatten.

Die erste Zielgruppe sind Surfer im Internet, die zufällig auf den Link zu [Methana 3D](#) stossen, sich die Halbinsel betrachten und Informationen abrufen. Dadurch wird bei ihnen Interesse an der Insel geweckt. Auch Personen, die gezielt im Internet nach Urlaubsmöglichkeiten suchen, können Gefallen an Methana finden und sich für einen Urlaub auf der Insel entscheiden. Die zweite Zielgruppe sind Urlauber, die schon wissen, dass sie auf die Insel fahren

werden und sich jetzt vorab genauer informieren wollen. Deshalb empfiehlt es sich, Themen wie Übernachtungsmöglichkeiten, Orte zur Freizeitgestaltung, Sehenswürdigkeiten und Ausflugsmöglichkeiten zu präsentieren.

Ein Informationssystem kann aber auch über nicht direkt touristische Themen Auskunft geben. Für Methana mit einer langen Geschichte bietet sich die Archäologie an mit Ort und Ausprägung von Fundstücken. Weiter hat die Insel, die durch Vulkane geschaffen wurde, eine interessante Geologie.

Viele Informationssysteme dieser Art beschränken sich auf zwei Dimensionen. Mit Methana 3D wird nun untersucht, wie die dritte Dimension im Internet präsentiert werden kann. Dem Betrachter soll eine leicht verständliche, grafische Benutzeroberfläche zur Verfügung gestellt werden, mit der er interaktiv Einfluss auf die virtuelle Welt nehmen kann.

Dazu muss zum einen ein geeignetes digitales Geländemodell (DGM) aufgebaut werden, zum anderen die vorhandenen Vektorkarten integriert werden. Die Arbeitsabläufe sollen weitgehend automatisch bzw.

halbautomatisch ablaufen, damit das Projekt mit wenig Aufwand auf ähnliche Informationssysteme von anderen Gebieten übertragen werden kann.

Einzelobjekte sind in einer Datenbank zu speichern. Sie werden mit Zusatzinformationen verlinkt, die ebenfalls in der Datenbank gespeichert sind.

1.4 Aufbau der Arbeit

Nach den Definitionen zu Methana 3D wollen wir uns einen Überblick über die zur Zeit erhältlichen online 3D-Visualisierungswerkzeuge verschaffen. Darauf folgt eine kleine Einführung zu VRML mit den Möglichkeiten, die es für die Kartografie bietet. Im Anschluss lernen wir die Halbinsel Methana kennen. Im nächsten Teil werden die in Methana 3D enthaltenen Objekte beschrieben und ausführlich darauf eingegangen, wie sie erzeugt wurden. Nach der Beschreibung der Datenbank wird die Funktionsweise von Methana 3D erklärt. Den Abschluss bildet eine Zusammenfassung der Probleme, die nicht gelöst werden konnten und ein Ausblick, wie Methana 3D erweitert werden kann.

2. Definitionen

In den folgenden Abschnitten wollen wir die einzelnen Begriffe einer kartografischen Online-3D-Plattform mit Datenbankanbindung definieren.

2.1 Kartografisch

G. HAKE & D. GRÜNREICH 1994, S. 3 definieren die *Kartografie* allgemein als „Fachgebiet, das sich befaßt mit dem Sammeln, Verarbeiten, Speichern und Auswerten raumbezogener Informationen sowie in besonderer Weise mit deren Veranschaulichung durch kartographische Darstellungen“, wobei die kartografischen Darstellungen (auch als kartografische Ausdrucksformen bekannt) vor allem durch „ein System geometrisch gebundener graphischer Zeichen aus einem endlichen, mit vereinbarten Bedeutungen versehenem Zeichenvorrat“ gekennzeichnet sind.

Unter *kartografischen Ausdrucksformen* versteht Arnberger „jene zwei- oder dreidimensionale Wiedergaben der Erdoberfläche, welche zur Wahrung der Lagebeziehungen nach vorgegebenen geometrischen Gesetzmäßigkeiten in einem bestimmten Verkleinerungsverhältnis nach kartographischen Prinzipien entworfen ist.“ (E. ARNBERGER 1993, S. 15)

Bei Methana 3D handelt es sich um eine kartografische Ausdrucksform, weil die geometrischen Gegebenheiten der Halbinsel in ein ebenfalls geometrisches Modell umgesetzt werden, in dem die Objekte mittels Signaturen in dreidimensionaler Form abgebildet werden.

2.2 Online

„On-line-Betrieb ist in der Datenverarbeitung eine Betriebsart, bei der Peripheriegeräte direkt an ein Computersystem angeschlossen sind, so daß der Entstehungsort von Daten u. der Rechner ständig verbunden sind.“ (BERTELSMANN UNIVERSALLEXIKON 1994)

Heute versteht man unter *online* meist das Übertragen von Daten übers Internet, wobei das Internet der weltweite Zusammenschluss von Rechnern und Netzen für den Datenaustausch und den Zugriff auf Datenbanken ist.

Bei Methana 3D sollen alle notwendigen Daten zentral auf einem Internetserver gespeichert werden, um

dann von Interessierten übers Internet geladen und am eigenen Rechner dargestellt zu werden.

2.3 Plattform

BROCKHAUS 1992, Bd. 17 S. 241, definiert *Plattform* als „an einem erhöhten Ort geschaffene ebene Fläche“. Eine solche Fläche ist gut einsehbar und eignet sich deshalb, um Informationen zu präsentieren. In dieser Arbeit wird Plattform als „Bühne“ (DUDEN BD. 8 1986, S. 513) verstanden, auf der Methana 3D dargestellt wird.

2.4 3D

3D ist eine „Abkürzung für *dreidimensional*“. Damit bezeichnen wir eine „Technik, um Bilder oder Objekte in drei räumlichen Dimensionen darzustellen oder Klänge in einer virtuellen Anwendung wirklichkeitsgetreu wiederzugeben.“ (F. FLUCKIGER 1996, S. 700)

Dabei ist zu beachten, dass die Darstellung immer noch auf einem zweidimensionalen Monitor geschieht. Aber durch eine entsprechende Bereitstellung der Daten und den Einsatz spezieller Software kann die Darstellung schnell geändert werden, um die Objekte aus einer anderen Richtung zu betrachten.

Methana 3D ermöglicht, dass sich der Betrachter frei auf der Halbinsel bewegen kann. Dazu muss der Rechner die notwendigen Darstellungen in *Real-Time* – also in Echtzeit – berechnen. Damit können wir den Begriff *virtuelle Realität (VR)* verwenden. Die moderne Vorstellung von VR als Anwendung, die dem Benutzer mit elektronischen Zusatzgeräten, z. B. einer stereoskopischen Brille oder einem Sensoranzug, den Eindruck der Immersion vermittelt, also dass er in die vom Computer dargestellte, künstliche Wirklichkeit eintaucht, ist auch bei Methana 3D denkbar, wurde aber nicht getestet.

2.5 Datenbankanbindung

Datenbankanbindung bedeutet, dass Informationen für eine Anwendung in einer Datenbank gespeichert und bei Bedarf abgerufen werden. Eine Datenbank ist „ein System aus Datengrossspeichern“. Es „enthält viele Daten eines Wissensgebiets, die schnell abgerufen werden können.“ (BERTELSMANN UNIVERSALLEXIKON 1994)

Der wichtigste Vorteil ist die einfache Möglichkeiten, Daten anzupassen und zu aktualisieren. Die Daten müssen nur einmal bearbeitet werden und alle Anwendungen, die darauf zugreifen, verwenden die neuen Daten.

Methana 3D bezieht einen Teil seiner Objekte aus einer Datenbank. Darauf gehen wir in einem späteren Kapitel näher ein.

2.6 Zusammenfassung

Eine kartografische Online-3D-Plattform mit Datenbankbindung ist also ein komplexes System, für das ein geografisches Gebiet angepasst und in räumlicher Form dargestellt wird, das die Inhalte aus einer Datenbank bezieht und mit dem der Betrachter via Internet interagieren kann.

3. Werkzeuge

3.1 Werkzeuge zur Real-Time-3D-Visualisierung

3.1.1 Eine kleine Markübersicht

In diesem Unterkapitel soll eine Übersicht gegeben werden, welche Programme und Formate zur 3D-Visualisierung im Internet zu finden sind (Stand September 1999). Grob kann man sie in drei Gruppen einteilen. Die erste Gruppe eignet sich hauptsächlich zum Betrachten von Einzelobjekten. Hier können wir ein Objekt untersuchen (examine), d. h. drehen, verschieben und zoomen. Verwendung findet

die Gruppe häufig im eCommerce, um die Produkte zu präsentieren, oder als animierte Banner mit kleinen Dateigrößen.

Die zweite Gruppe ermöglicht ein komfortables ‚Eintauchen‘ in die Objekte. Das heisst zusätzlich zu den Untersuchungsfunktionen können wir auch fliegen, gehen und uns neigen. Damit eignet sie sich, um durch Räume und Landschaften zu fliegen oder auch zur Visualisierung komplexer Daten.

Die letzte Gruppe ist dann die Erweiterung zur Multi-User-Welt. Hier können gleichzeitig mehrere Personen in der Welt unterwegs sein. Wir können die Bewegungen der anderen Besucher sehen und miteinander kommunizieren.

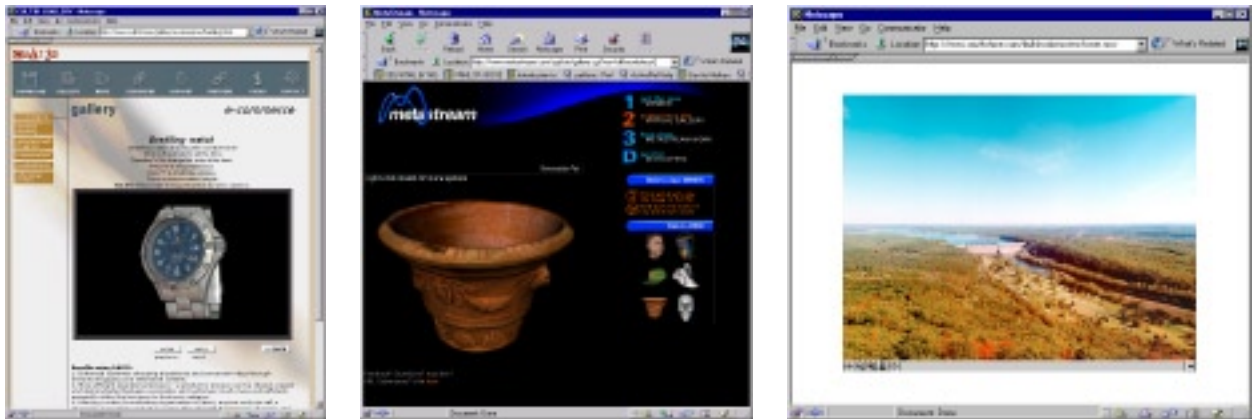


Abbildung 2: Werkzeuge zur 3D Objektbetrachtung im Internet (v. l.): Cult 3D, MetaStream, QuickTime.

3.1.2 Werkzeuge zur Objektbetrachtung

(Abb. 2)

3D Dreams 2.0 von Shells Interactive
Integriert die dritte Dimension in Director und kann mit Shockwave auf Webseiten betrachtet werden. In erster Linie zur Objektbetrachtung. Interaktion möglich, Sound, Animation.
<http://www.DoItin3D.com/>

Cult 3D von Cycore
In erster Linie zum Betrachten von Objekten. Erzeugen der Geometrie mit 3D Studio MAX, Interaktivität mit Cult 3D Designer hinzufügen, Sound möglich, schwedische Firma.
<http://www.cult3d.com/>

MetaStream™ von MetaCreations® Corporation
Offenes PC-Dateiformat, nur zum Betrachten von Objekten.
<http://www.metastream.com/>

QuickTime von Apple Computer, Inc.
Gehört eigentlich nicht zu den Real-Time-Anwendungen. Hier können Panoramen oder Einzelobjekte betrachtet werden, von denen der Ersteller Rasterbilder ins Web gestellt hat. Diese können miteinander verlinkt werden, so dass man auch auf die Reise in einer virtuellen Realität gehen kann. Diese Technik basiert auf Rastergrafiken, die auf die Oberfläche, bzw Innenfläche einer Kugel projiziert werden. Durch die vorgerechneten Bilder kann eine hohe Qualität auf Kosten der Interaktivität erzielt werden, grosse Dateigrößen.
<http://www.apple.com/quicktime/>

SmoothMove von iMove, Inc.
Ähnlich Quicktime, nur nicht so verbreitet.
<http://www.smoothmove.com/>

3.1.3 Werkzeuge für Single-User-Welten

(Abb. 3 - 5)

3DML™ von Flatland Online, Inc.

Eine eigene Websprache basierend auf Tags (wie HTML) zum Erzeugen von dreidimensionalen Welten. Durchfliegen der Welten möglich.
<http://www.flatland.com/>



Abbildung 3: Screenshot von 3DML.

Alice Interactive 3D Graics Programming System

von Carnegie Mellon University
Einfache Programmiersprache, Objektbetrachtung und Durchfliegen von Welten möglich, an Universitäten entwickelt.
<http://www.alice.org/>

Hypercosm von Hypercosm, Inc.

Programmiersprache, kleine Dateigrößen
<http://www.hypercosm.com/>



Abbildung 4: Screenshot von Shout3D.

Java3D von Sun Microsystems, Inc.

Programmier-API für Java-Anwendungen, die ihrerseits animierte Inhalte präsentieren. Anwendungen laufen auf jedem Java-2-fähigen Betriebssystem. Durch Java kaum Einschränkungen für Interaktion oder Multi-User-Welten.

<http://java.sun.com/products/java-media/3D/index.html>

Shout3D von Shout Interactive, Inc.

In Java geschrieben, kann ohne Plug-in von jedem Java-1.02-fähigen Browser angezeigt werden. Shout Interactives Vorschlag für X3D.

<http://www.shout3d.com/about.html>

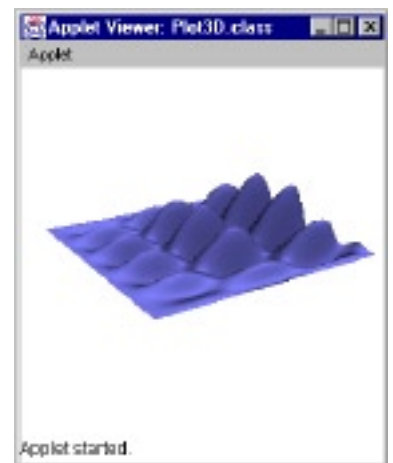


Abbildung 5: Screenshot von Java3D.

VRML97 vom Web3D Consortium, Inc.

Virtual Reality Modeling Language, verbreitetste Beschreibungssprache für 3D-Welten, International Standard ISO/IEC 14772, durch verschiedene Skriptmöglichkeiten kann gut auf die Welten Einfluss genommen werden, Erweiterung zur Multi-User-Welt möglich.

<http://www.web3d.org/>

X3D vom Web3D Consortium, Inc.

Extensible 3D, nächste Generation von VRML, im Aufbau. Web3D „hat sich zum Ziel gesetzt, einen Set schlanker Komponenten zu spezifizieren, die sich je nach den Anforderungen einer spezifischen Anwendung kombinieren lassen.“ (X3D 1999) Die ersten Komponenten sollen ein 3D-Runtime-Engine mit Rendering-Funktionen, ein plattformunabhängiges Dateiformat und die XML-Integration sein.

http://www.web3d.org/fs_technicalinfo.htm

3.1.4 Werkzeuge für Multi-User-Welten

Active Worlds von Activeworlds.com, Inc. (Abb. 6)
Eigenständiges Programm für Multi-User-Welten
<http://www.activeworlds.com/>

blaxxun Contact 4.2 von blaxxun interactive
3D-Mehrbenutzerumgebung auf Basis von VRML.
Einer der beliebtesten 3D-Chats ist sicher Colony
City/Cybertown, München, San Francisco.
<http://www.blaxxun.com/>



Abbildung 6: Screenshot von Active Worlds.

Entrance von CyberCore Systems (Abb. 7)
3D-Mehrbenutzerumgebung in Web-Browser, hohe
Datenkompression, Firmensitz in Bochum
<http://www.CyCoSys.com/entrance/index.html>

3.1.5 Werkzeugfestlegung Real-Time-3D-Visualisierung

Um eine geeignete Technik für Methana 3D zu finden, wurden mehrere Gesichtspunkte betrachtet. Die zu übertragende Datenmenge soll handlich bleiben, damit der Besuch von Methana 3D nicht durch lange Ladezeiten erschwert wird. Hier schneiden QuickTime und ähnliche Werkzeuge trotz ihrer hervorragenden Qualität schlecht ab. Weiter soll es dem Betrachter bei Methana 3D möglich sein, wie mit einem Flugzeug über die Insel zu fliegen und sie aus jeder beliebigen Richtung zu betrachten. Damit scheiden alle Werkzeuge, die primär zur Objektbetrachtung entwickelt wurden, aus. Dann soll der Betrachter Informationen zu- und abschalten können. Es muss

also gute Möglichkeiten zur Steuerung der Objekte geben. Schliesslich soll Methana 3D möglichst schnell für möglichst viele Internetnutzer erreichbar sein. Deshalb scheiden die Techniken, die noch in Entwicklung sind wie Java3D oder X3D aus. Weil es nicht nötig ist, dass sich mehrere Besucher der Insel gegenseitig sehen, braucht es keine Multi-User-Welt zu sein.

Bei VRML treffen alle Anforderungen weitgehend zu. Aber auch hier müssen Nachteile in Kauf genommen werden. VRML lässt sich im Internet-Browser nur mit Plug-in betrachten. Diese sind selbst für VRML nicht sehr verbreitet. Wie viele Besucher sich die Mühe machen werden, die nicht gerade kleinen Plug-ins zu laden und zu installieren, wird sich zeigen. Immer wieder bereitet es Schwierigkeiten, dass verschiedene VRML-Viewer den Code unterschiedlich darstellen. Schlimmer als beim Kampf zwischen Microsoft Internet Explorer und Netscape um HTML Befehle scheint es fast unmöglich, komplexe Welten für mehrere Plug-ins zu entwickeln. Allerdings sind hierfür nicht marktstrategische Gesichtspunkte die Ursache, sondern unterschiedlicher Ressourcen-Einsatz bei den entwickelnden Firmen und unterschiedliche Interpretation der VRML-Spezifikation.

Weil VRML in ASCII beschrieben wird, sind die Datenmengen nicht so klein, wie sie in einem binären Format sein könnten. Doch lassen sich VRML-Dateien mit GZIP komprimieren und damit auf ungefähr ein Viertel verkleinern. Leider hat sich der Durchbruch für diesen 3D-Standard immer noch nicht eingestellt, und früher sehr aktive Firmen wie Platinum entwickeln nicht mehr für VRML. Doch soll VRML vollständig in den Nachfolgestandard X3D integriert werden. Hoffnung machen auch Projekte wie z. B. die Entwicklung von Java Applets, die VRML ohne Plug-in darstellen können.
(<http://www.blaxxun.com/products/blaxxun3d/index.html>)

Für VRML spricht, dass es sich hervorragend mit Programmiersprachen beeinflussen lässt. Zum einen kann die Welt durch einen *Script-Knoten* von sich aus komplexe Aktionen ausführen. Zum anderen kann über das *External Authoring Interface* (EAI) die Welt

auch von aussen beeinflusst werden. Inzwischen ist ein EAI-Vorschlag als Erweiterung des VRML-Standards bei der ISO in Bearbeitung, ist aber nur bei wenigen Plug-ins implementiert. Die wichtigsten Programmiersprachen für VRML sind Javascript und Java.

Durch den offenen Standard eignet sich VRML auch gut zur Entwicklung eigener Welten, weil man nicht von Entwicklungsumgebungen abhängig ist und genau dokumentiert ist, wie etwas zu implementieren ist.

Für Methana 3D ist VRML also ein umfangreiches und geeignetes Werkzeug und wird deshalb für die 3D-Visualisierung verwendet.



Abbildung 7: Screenshot von Entrance.

3.2 Werkzeuge zur Datenbearbeitung

Pixel PRO 01.01.00.00 von Intergraph Corporation

Zur Konvertierung von RLE-Dateien in TIFF-Dateien.

Microstation 05.00.95.48 von Bentley Systems, Incorporated / Intergraph Corporation

Zur Konvertierung der Ausgangsvektoren der Karten aus dem DGN-Format ins DXF-Format.

Arc/Info 7.2.1 von Environmental Systems Research Institute, Inc. (Esri)

Das GIS Paket wurde zur Bearbeitung des digitalen Geländemodells (DGM), zum Bearbeiten des Vektorkartenmaterials, zum Digitalisieren und zur Verschneidung von Objekten mit dem DGM verwendet.

Perl 5.005_02 von Larry Wall

Perl-Interpreter, für ASCII-Dateikonvertierung verwendet, CGI-Script-Sprache, Online-Abfrage an Datenbank (DBI.pm-/ DBD::mysql.pm-Module), Erzeugen von VRML-Objekten (VRML-Perl-Modul).

mysql 9.32

Multiuser-SQL-Datenbankserver. In der Datenbank werden Informationen zu Häusern oder Sehenswür-

digkeiten gespeichert, die von Methana 3D abgerufen werden können.

UltraEdit 6.00c von IDM Computer Solutions, Inc. Text- und HEX-Editor mit „Syntax Highlighting“ zum Programmieren in allen Programmiersprachen.

gzip 1.2.4 (GNU zip)

Zum Komprimieren der VRML Dateien.

IrfanView 3.02 von Irfan Skiljan, Österreich

Freeware Bildbetrachter für Rastergrafiken, Erzeugen von Thumbnails, Batchkonvertierung. Geheimtipp für alle, die Fotografien im Internet präsentieren wollen. Mit der Batchkonvertierung ist es möglich, ganze Ordner von Rasterdateien auf einmal zu verkleinern, zu schärfen und in ein beliebiges Format zu speichern. (<http://stud1.tuwien.ac.at/~e9227474/>)

Photoshop 5.0 von Adobe

Das Bildbearbeitungsprogramm wurde zur Erstellung aller Rastergrafiken, zur Kombination von Rastergrafiken und zur Dokumentation verwendet.

Freehand 8.0 von Macromedia

Vektorgrafikprogramm zur Erzeugung und Überprüfung von Grafiken.

Internet Explorer von Microsoft
Hauptsächlich für Online-Dokumentationen und
Online-Wörterbücher.

Communicator 4.5 von Netscape Communications
Corporation
Als Entwicklungsumgebung von Methana 3D und
zur Online-Telekommunikation.

Word 97 von Microsoft
Textverarbeitung.

Indesign von Adobe
Layout.

Windows NT 4.00 von Microsoft
Betriebssystem.

Linux & Apache
Server-Plattform.

4. Einführung in VRML

4.1 Entstehung

Im Mai 1994 stellten Mark Pesce und Tony Parisi den Prototyp eines 3D-Browsers auf der WWW-Konferenz in Genf vor. Das sich daraus entwickelnde Dateiformat VRML 1.0 (Virtual Reality Markup Language) wurde weitgehend vom Open-Inventor-Dateiformat von SGI (Silicon Graphics Inc.) abgeleitet. Während bei den HTML-Browsern die Entwickler den Standard quasi bestimmen, wurde bei VRML zuerst die Spezifikation festgelegt und danach die Software (Browser) programmiert. Im April 1995 wurde die VRML-1.0-Spezifikation herausgebracht, die nur einfache, statische 3D-Welten vorsah.

Auf der Siggraph-Konferenz im August 1996 wurde die VRML-2.0-Spezifikation freigegeben. Mit dieser Version wurde es möglich dynamische Welten zu generieren. Mit einem event-orientierten Modell und über Skripte sind Animationen steuerbar und kann auf Aktionen des Betrachters reagiert werden. Auch die Schnittstelle zur Aussenwelt wurde definiert. Im September 1997 wurde die Spezifikation unter dem Namen VRML97 zum International Standard 14772 der International Standards Organization (ISO) erklärt.

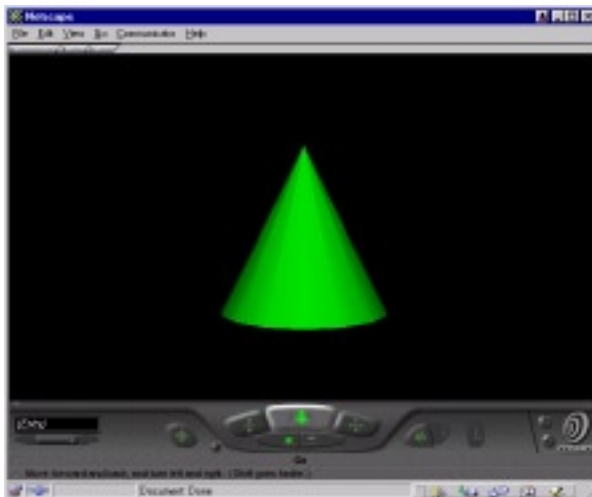


Abbildung 8: Grüner Kegel in VRML.

Das Web3D-Konsortium (früher VRML-Konsortium) hat sich zur Aufgabe gemacht, den Standard zu pflegen und die Weiterentwicklung zu fördern.

4.2 Aufbau

VRML-Dokumente sind Dateien aus ASCII-Text, die systemunabhängig mit einem Texteditor geschrieben werden können. Sie haben die *Extension* „.wrl“.

Alle Objekte werden durch Knoten beschrieben. Mit Geometrieknoten definiert man z. B. Kugeln oder Kegel, mit Materialknoten die Eigenschaften des Materials, z. B. Farbe oder Transparenz. Gruppenknoten ordnen die Objekte hierarchisch. Eine solche Hierarchie wird Szenegraph genannt. Der Szenegraph hat eine Baumstruktur, in dem die Gruppenknoten auf untergeordnete Knoten verweisen. Material- oder Geometrieknoten sind sogenannte Blattknoten. (O. SCHLÜTER 1998, S. 35)

Den Aufbau des Szenegraph kann man sich ähnlich einem Bausatz eines Modellflugzeugs vorstellen. Zuerst hat man die Einzelteile (Blattknoten) wie Propeller, Tragflächen, Flugzeugrumpf. Diese werden zu einzelnen Gruppen zusammengebaut, z. B. Flügel. Am Schluss werden diese Untergruppen mit dem Rumpf zum fertigen Flugzeug zusammengesetzt. (D. NADEAU 1999, S. 26f)

Ein einfaches Beispiel verdeutlicht den Aufbau einer VRML Datei (Abb. 8):

```
01 #VRML V2.0 utf8
02 Shape {
03     appearance Appearance {
04         material Material {
05             diffuseColor 0 1 0
06         }
07     }
08     geometry Cone {
09         height 3.0
10         bottomRadius 1.5
11     }
12 }
```

Zeile 01 beschreibt den Header jeder VRML97 Datei.

Shape {} (Zeile 02) bzw. Cone {} (Zeile 08) beschreiben einen Knoten. Jeder Knoten besitzt ein Paar geschweifte Klammern und null oder mehr Felder innerhalb der Klammern, die die Attribute des Knotens beinhalten.

Der Kegel Cone {} ist 3.0 Einheiten hoch und hat einen Radius von 1.5.

Die Form Shape {} hat z. B. das Attribut appearance (Zeile 03). Darin wird das Material des Kegel festgelegt. Der Kegel ist grün (Zeile 05).

Auf eine ausführliche Beschreibung von VRML z. B. der einzelnen Knoten, wird hier verzichtet. Der interessierte Leser kann sich in jedem VRML-Lehrbuch informieren. Wir werden jetzt nur drei für Methana 3D wichtige Charakteristika ausführlicher betrachten.

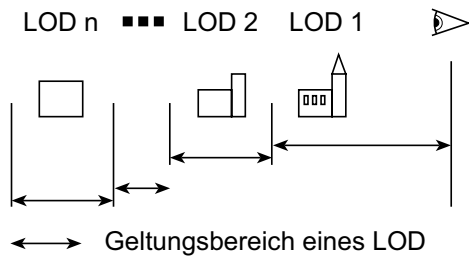


Abbildung 9: Das LOD Prinzip (Nach: M. SUTER 1997, S. 60).

4.3 Wichtige VRML-Charakteristika für Methana 3D

Aus dem Alltag ist uns bekannt, dass wir entfernte Objekte durch die perspektivische Verkleinerung nicht so genau erkennen können wie nahe. Also brauchen wir in der virtuellen Realität für Entferntes eine geringere Detailtreue als für Nahes. Einem Objekt unterschiedliche Geometrie in Abhängigkeit von der Entfernung zum Betrachter zuzuweisen (Abb. 9), wird in der Computergrafik *Level of Detail* (LOD) genannt. Weil einfachere Geometrie schneller zu rechnen ist, erhöht sich die Darstellungsgeschwindigkeit. VRML bietet LOD-Knoten, welche vor allem in komplexen Welten zum Einsatz kommen.

Insgesamt existieren im VRML-Standard 54 Knoten. Zusätzlich zu diesen können auch mit Hilfe von Prototypen eigene Knoten definiert werden, die dann beliebig oft wiederverwendbar sind. Prototypen haben eigene Eigenschaften und Fähigkeiten. Sie eignen sich vor allem für Objekte, die häufig in gleicher oder ähnlicher Form vorkommen.

Mit dem *External Authoring Interface* (EAI) (EAI 1999) ist es möglich, den Inhalt eines VRML-Plug-in-Fensters, das in eine Webseite eingebettet ist, von ausserhalb zu steuern. Programmiersprachen wie z. B. Javascript oder Java lassen mit Hilfe von Live-Connect bzw. ActiveX/COM auf die Plug-ins des

WWW-Browsers zugreifen. Im Fall von Netscape/LiveConnect stellt jedes VRML-EAI-Plug-in Java-Klassen zur Verfügung, um neue VRML-Objekte zu erstellen, zu beeinflussen oder sie zu löschen (Abb. 10). Für die Interaktion mit dem Benutzer kann man HTML-Formulare verwenden. Ausserdem ist es möglich, Informationen aus der VRML-Welt auszulesen, z. B. die aktuelle Position des Besuchers festzustellen. Multi-User-Welten, in denen sich mehrere Personen aufhalten und sich gegenseitig sehen, benötigen ebenfalls das EAI.

Wegen ihrer Komplexität beschreiben wir jetzt die Steuerung, wie zusätzliche VRML-Szenen in eine geladene Welt einzubauen sind:

```

01 DEF SIGNS Group {}
02 DEF CREATESIGNS Script {
03     eventIn SFString newurl
04     field SFNode signsGroup USE SIGNS
05     url ["vrmlscript:
06         function newurl(value) {
07             myUrl = new MFString(value);
08             Browser.createVrmlFromURL(myUrl,
09                 signsGroup, 'addChildren');
10         }
11     "
12 ]
13 }
    
```

Wir benötigen zwei Knoten. Der erste ist ein Gruppenknoten, in den die neue Geometrie eingebaut wird (Zeile 01). Der zweite ist ein Skriptknoten, der die Steuerung übernimmt (Zeile 02 - 13).

Das Skript hat zwei Felder. Das *eventIn* (Zeile 03) löst die Funktion *newurl* (Zeile 06 - 10) aus. Zeile 04 ist eine Referenz auf den Gruppenknoten.

Bekommt das Skript als Ereignis (*eventIn*) die URL der einzubauenden Datei, wird die Funktion *newurl* ausgeführt. Ihr Parameter (*value*) ist die URL. Weil die Funktion *createVrmlFromURL* ein Feld von URLs braucht und nicht nur einen einzelnen, legen wir ein

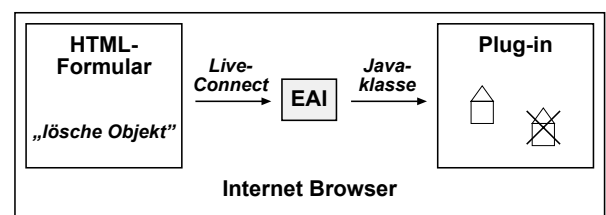


Abbildung 10: Informationsfluss über das EAI.

solches Feld an (Zeile 07). Es beinhaltet nur die übergebene URL (*value*).

Jetzt können wir die Funktion *createVrmlFromURL* mit drei Parametern aufrufen (Zeile 08, 09):

- *myUrl* ist die URL, von der die VRML Datei gelesen wird.
- *signsGroup* ist der Knoten, an den das Ereignis *addChildren* gesendet wird.
- *addChildren* enthält die VRML-Szene.

Die VRML-Szene aus *myUrl* wird also dem Gruppenknoten *signsGroup* durch das Ereignis *addChildren* angehängt.

Die URL muss nicht zwingend eine bestehende VRML-Datei beinhalten, sondern kann auch ein CGI-Skript aufrufen, dass den VRML Inhalt erzeugt.

4.4 Zukunft VRML

VRML ist starker Kritik ausgesetzt. Die Plug-ins sind mit über 2 MB zu gross. Trotz des ISO-Standards gibt es deutliche Unterschiede zwischen den Plug-ins. Optimal lässt sich eine Welt gewöhnlich nur mit dem Plug-in betrachten, mit dem es entwickelt wurde. Oft sind die Plug-ins instabil, mit Abstürzen muss gerechnet werden. Das alles hat verhindert, dass VRML für ein breites Publikum interessant wurde. Der erwartete Durchbruch ist immer noch nicht absehbar. Als im Frühjahr 1999 Platinum, der Besitzer des verbreitetsten Plug-in Cosmoplayer, die komplette VRML-Abteilung entliess und mitteilte, dass Cosmoplayer nicht mehr weiter entwickelt wird, schien das Aus für VRML besiegelt.

Doch die jüngsten Entwicklungen ermutigen dazu, sich weiterhin mit 3D im Internet zu beschäftigen. Es gibt inzwischen mehrere Alternativen zum Cosmoplayer. Beispielhaft seien Blaxxun Contact und ParallelGraphics Cortona VRML Client genannt.

Zukunftsweisend sind Projekte von Shout Interactive und Blaxxun. Beide haben Plattformen programmiert, die ohne Plug-in auskommen. Sie sind in Java programmiert und können von jedem javafähigen Browser dargestellt werden.

Das wohl meist gehörte Schlagwort im Zusammenhang von 3D und Internet ist sicher Java3D. Es wurde von Sun Microsystems entwickelt. Java3D ist eine 3D-Szenegraph-basierte Programmierschnittstelle (API) für Java und unterstützt Routinen zur Erzeugung von 3D-Geometrie. Um Java3D ausführen zu können, muss es kompiliert werden. Ausserdem benötigt man Java 2. Das bedeutet, dass Applets in den aktuellen Browsern (Herbst '99) nur mit Hilfe eines Plug-ins laufen. VRML-Welten können in Java3D eingebunden werden.

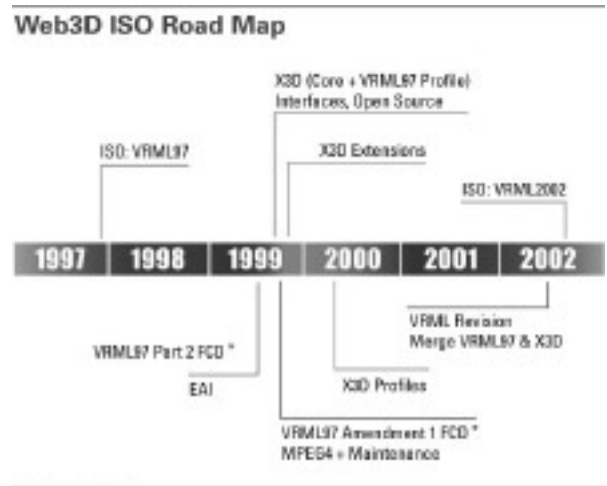


Abbildung 11: Road Map von Web3D.

Das Web3D-Konsortium hat sich für das Jahr 2002 zum Ziel gesetzt, einen neuen VRML-Standard zu definieren (Abb. 11). Dazu entwickelt eine Arbeitsgruppe Extensible 3D (X3D). X3D soll die Geometrie und das Verhalten von VRML97 mit Hilfe der Extensible Markup Language (XML) ausdrücken. Die Arbeitsgruppe arbeitet eng mit dem World-Wide-Web-Konsortium (W3C) zusammen, um 3D in die nächste Generation von Web-Standards zu integrieren. X3D soll aus XML-Tags bestehen, einzelne Komponenten besitzen, erweiterbar und abwärtskompatibel sein. Die ersten X3D-Applikationen sollen im ersten Halbjahr 2000 lauffähig sein. (X3D 1999)

Nach dieser Übersicht zu VRML wenden wir uns jetzt den Anwendungsmöglichkeiten in der Kartografie zu.

5. VRML in der Kartografie

5.1 Virtual Reality (VR)

VRML ist Dank seines einfachen Aufbaus, seiner vielfältigen Möglichkeiten und seines grossen Potentials zur schnellen Verbreitung zu einem gängigen Mittel geworden, um künstliche Welten von geografischen Sachverhalten zu erstellen. Eine virtuelle Realität kann uns helfen zu verstehen, wo wir uns im Raum befinden und reflektiert die reale Welt in einer intuitiven Weise. Interaktivität und Dynamik kann das Engagement des Betrachters und dessen Verständnis fördern. Damit wird VRML zu einem neuen Werkzeug des Kartografen, mit dem er räumliche Sachverhalte modellieren kann (K. MOORE 1999, S. 205f).

In der VR kann das „Lesen“ der Karten erfahrbar werden. Zum einen erleben wir eine Karte, in die man quasi eintauchen kann (Immersion), intensiver als Karten, die sich auf zwei Dimensionen beschränken. Zum anderen können wir an der Karte teilnehmen, wenn wir unseren Blickwinkel einstellen und sie nach unseren Wünschen gestalten können. Deshalb ist es wichtig, dem Betrachter die Möglichkeit zu geben, Objekte an- und abzuschalten, Abfragen zu machen oder aus verschiedenen Themen auszuwählen. Damit verschwimmt auch die Grenze zwischen topografischer und thematischer Karte, weil je nach Wunsch entsprechende Objekte angezeigt werden. (J. SWANSON 1999, S. 181 - 184)

DYKES et al. 1999 weisen darauf hin, dass mit VRML die Realität einerseits noch besser abgebildet werden kann als z. B. mit Imhofs Landschaftsgemälden, aber andererseits auch sehr abstrakte Sachverhalte in einer realen Welt visualisiert werden können. Das ist z. B. der Fall, wenn die Gebäude einer Stadt nach ihrer Nutzung eingefärbt oder wenn statistische Daten über ein Gelände projiziert werden.

5.2 Der praktische Gebrauch

In der Literatur wird auf verschiedene Punkte hingewiesen, die es bei dem Aufbau einer kartografischen VRML-Welt zu beachten gilt. Der «Spielfluss» des Betrachters sollte nie unterbrochen werden. Das bedeutet, dass wir Ladezeiten und Rendering so effektiv wie möglich gestalten müssen. Das kann z. B. heissen, dass wir Texturen statt komplizierten Geometrien verwenden oder dass wir viele LODs einsetzen. Nur kontrastreiche Farben lassen sich unter den verschiedenen Beleuchtungen richtig identifizieren.

Statische Karten können wir durch Hyperlinks und Multimediaelemente interessanter gestalten. Vor allem dem Ton ist in der Kartografie nur wenig Beachtung geschenkt worden. Eine 2D-Karte mit der aktuellen Position hilft dem Benutzer sich zu orientieren und vermeidet, dass er sich in der VR „verliert“. In eine



Abbildung 12: Screenshot von «Glasgow Directory».

georeferenzierte VRML Welt können Objekte aus einer geografischen Datenbank integriert werden. Ein weiterer Aspekt sind Multi-User-Welten, die der realen Welt noch näher kommt, da wir uns dort ebenfalls nicht alleine befinden.

Besondere Beachtung müssen wir einer Anleitung oder Hilfe schenken, damit der Betrachter die vielfältigen Möglichkeiten, die ihm geboten werden, auch nutzen kann. (J. A. DYKES et al. 1999, K. MOORE 1999, S. 205 – 216, J. SWANSON 1999, S. 192f)

Neben den Impulsen, die die Kartografie durch VRML bekommt, kann sie auch Impulse zurückgeben. Das soll mit Hilfe der GeoVRML Working Group möglich sein.



Abbildung 13: Platzierung von lokalen VRML-Welten auf dem Globus mit einem geoLocation-Knoten.

5.3 GeoVRML Working Group

Ziel dieser offiziellen Arbeitsgruppe des VRML-Konsortiums ist, zu diskutieren, wie georeferenzierte, dreidimensionale Daten in VRML vertreten sein können. Die Arbeitsgruppe soll Werkzeuge entwickeln, Techniken empfehlen und mögliche Verbesserungen oder Erweiterungen des VRML-Standards vorschlagen, um solche Daten zu erzeugen, darzustellen und auszutauschen.

Nach Auffassung der Arbeitsgruppe ist VRML ungeeignet, um Geodaten sowohl exakt zu repräsentieren als auch einfach zu erzeugen und zu verändern. GeoVRML will versuchen, die Informationen zu erkennen, die besonders nützlich sind, um die Genauigkeit beliebiger georeferenzierter Daten sicherzustellen. Weiter will die Arbeitsgruppe Methoden empfehlen, wie solche externen Daten in VRML-Datensätzen zur Verfügung gestellt werden können. Sie wollen Mittel vorschlagen, um Unzulänglichkeiten des gegenwärtigen Standards zu überwinden bzw. um den Standard in Zukunft zu verbessern.

Folgende Themen sind in Bearbeitung:

- Koordinatensysteme (verschiedene Koordinatensysteme, z. B. UTM, Definition des Ursprungs; Problem:

- VRML Datentyp Float nicht ausreichend für weltweit georeferenzierte Daten)
- Referenzzeit (coordinated Universal Time; UTC)
- Geländedarstellung (Problem: Datenmenge, grosse Bilder, umfangreiche Geländemodelle)
- Level of Detail (um grosse Datenmengen in sinnvoller Auflösung zu handhaben; viele Hierarchien)
- Auflösung und Genauigkeit (für Weltkoordinaten VRML Float nicht ausreichend; nur lokale Koordinatensysteme möglich)
- Datenaustausch (Zusammenfügen von verschiedenen Modellen, Metadaten)

Bis Ende 1999 soll ein Vorschlag zu GeoVRML 1.0, einer Erweiterung zu VRML97, erarbeitet sein (M. REDDY & L. IVERSON 1999, S. 30 - 38 und GEOVRML 1999).

Leider beschäftigt sich GeoVRML hauptsächlich mit der statischen Visualisierung von Geodaten und setzt die Möglichkeiten, die Interaktivität und Animation bieten, nicht ein.

5.4 GeoVRML 1.0

GeoVRML 1.0 (GEOVRML 1.0 DRAFT 1999) stellt eine Reihe von Lösungen zur Verfügung, um geografische Daten mit einem Standard-VRML97 Browser zu visualisieren. Dazu sind einige neue Knoten definiert, die sich die Erweiterbarkeit von VRML mit Prototypen zu Nutze machen. Für jeden GeoVRML-1.0-Knoten ist eine Open Source Beispiel-Implementierung vorhanden.

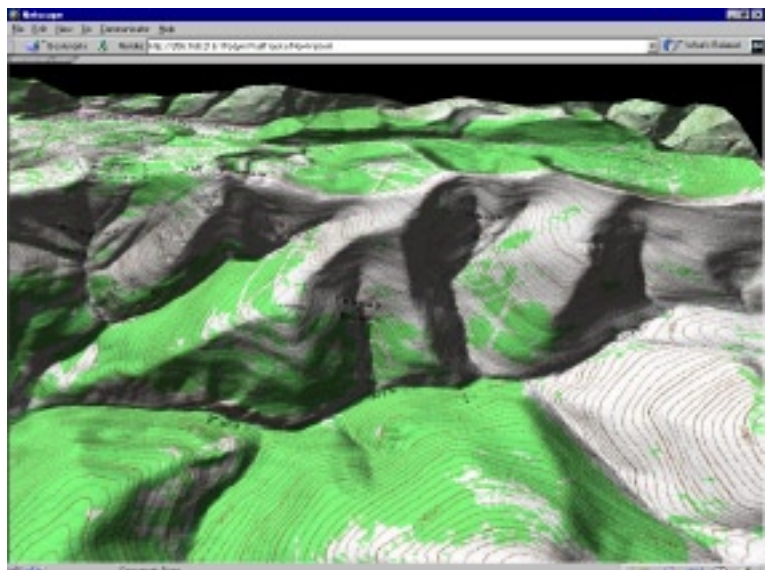


Abbildung 14: Gelände, das mit einer topografischen Karte als Textur belegt ist.

Folgende Gesichtspunkte wurden beim Vorschlag für GeoVRML 1.0 besonders beachtet:

Georeferenzierung soll weltweit eindeutig möglich sein. Dazu sollen verschiedene *Koordinatensysteme* verwendet werden können. Daraus ergibt sich das Problem, dass die *Genauigkeit* des Datentyps VRML-Single-Float nicht ausreicht.

Die grosse Datenmenge, die z. B. für die Darstellung eines ganzen Staates mit Höhenmodell und Textur nötig ist, kann nur schlecht im Internet verwendet werden. Deshalb soll die Gesamtfläche in einzelne Kacheln aufgeteilt werden mit jeweils eigenen *LOD*. Nur die sichtbaren Daten müssen geladen werden.

Im aktuellen Vorschlag wurden neun neue Knoten definiert:

- GeoCoordinate* Erstellt Geometrie aus geografischen Koordinaten
- GeoElevationGrid* Definiert ein DEM aus geografischen Koordinaten
- GeoInline* Bindet eine Datei ein mit der Kontrolle über den Lade- und Entladezeitpunkt der Daten
- GeoLocation* Platziert eine Standard-VRML-Welt georeferenziert auf der Oberfläche der Erde
- GeoLOD* *Level of Detail*, Verwaltung für unterschiedlich aufgelöstes Gelände
- GeoMetadata* Beschreibung der verwendeten geografische Daten
- GeoOrigin* Definiert ein lokales Koordinatensystem
- GeoPositionInterpolator* Animiert Objekte im geografischen Koordinatensystem
- GeoViewpoint* Definiert Kamerapositionen in geografischen Koordinaten

Beispielhaft wollen wir den *GeoLocation*-Knoten genauer betrachten.

```
EXTERNPROTO GeoLocation [
  field SFNode    geoOrigin    # NULL
  field MFString  geoSystem    # [ "GDC" ]
  field SFString  geoCoords    # ""
  field MFNode   children     # []
] "http://www.ai.sri.com/geovrml/1.0/
protos/GeoLocation.wrl"
```

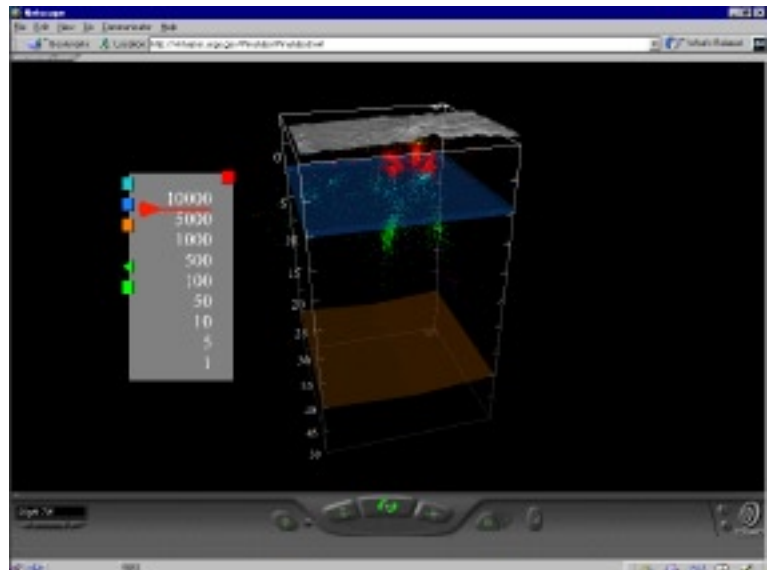


Abbildung 15: Visualisierung von Geodaten mit VRML (USGS).

Mit Hilfe dieses Knotens kann eine in normalen VRML-Koordinaten beschriebene Welt georeferenziert auf die Erdoberfläche platziert werden. Im Feld *children* ist die eingefügte Welt zu definieren, relativ zum Ursprung, der mit *geoCoords* spezifiziert ist.

geoSystem wird verwendet, um das Koordinatensystem zu bestimmen, in dem alle geografischen Koordinaten im Knoten beschrieben sind.

Mit *geoOrigin* lässt sich ein lokaler Nullpunkt ausweisen, auf den sich alle geografischen Koordinaten des Knotens beziehen.

Abbildung 13 zeigt uns den Einsatz eines solchen Knotens.

Beispielwelten, in denen Knoten nach GeoVRML 1.0 zur Anwendung kommen, finden wir auf <http://www.ai.sri.com/VRMLSets/>.

5.5 Beispiele

Die Mehrzahl der VRML Welten mit geografischem Bezug sind Welten von Städte oder Stadtteilen. Drei Beispiele wollen wir uns ansehen. Das Beispiel aus Helsinki ist wegen seines Umfangs und seiner geometrisch genauen Gebäude hervorzuheben (<http://www.arenanet.fi/virtuaalihelsinki/>). Das zweite Beispiel aus Strassburg ist interessant, weil sowohl die Orientierung als auch die Position im Fenster eingeblendet sind (<http://www.3dvr.org/vrstrasbourg/welcome1.html>). Das dritte Beispiel zeigt

ein umfangreiches Informationssystem von Glasgow, das *Glasgow Directory*. Themen sind individuell zu- und abschaltbar. Viele Objekte sind mit weiteren Information aus Multimediadatenbanken verlinkt (G. ENNIS et al. 1999). Übersichtskarten mit der Position des Besuchers und Multi-User-Welten sind in Vorbereitung (<http://iris.abacus.strath.ac.uk/glasgow/>, Abb. 12).

Ein weiterer grosser Bereich sind Geländemodelle, die mit einer Textur überlagert sind. Meist werden Satellitenbilder als Textur verwendet. Aber es finden sich auch Beispiele mit topografischen Karten oder thematischen Inhalten (<http://www.mtop.com/>, Abb. 14). Leider wird in manchen der Welten die Immersion dadurch behindert, dass nicht beachtet wurde, dass in VRML y- und z-Koordinate vertauscht werden müssen. Ein Durchfliegen einer Welt, die durch falsche Koordinaten senkrecht gestellt ist, ist nicht mehr in richtiger Lage möglich. Um die grosse Datenmenge für das Gelände und die Texturen über das Internet handhaben zu können, wird viel Aufwand für LOD und dynamisches Nachladen der benötigten Daten betrieben (M. REDDY, Y. LECLERC et al. 1999). Die meisten dieser Welten nutzen aber die Möglichkeiten des Mediums VRML zur Interaktion nicht aus (J. SWANSON 1999, S. 186).

Wirkungsvoller wird VMRL beim *traVelleR* (<http://www.geog.le.ac.uk/vfc/>, Abb. 16) eingesetzt, der an der University of Leicester entwickelt wurde. Auf einer 2D-Karte sehen wir unsere Bewegung in der 3D-Welt. Wir können aus verschiedenen Textu-

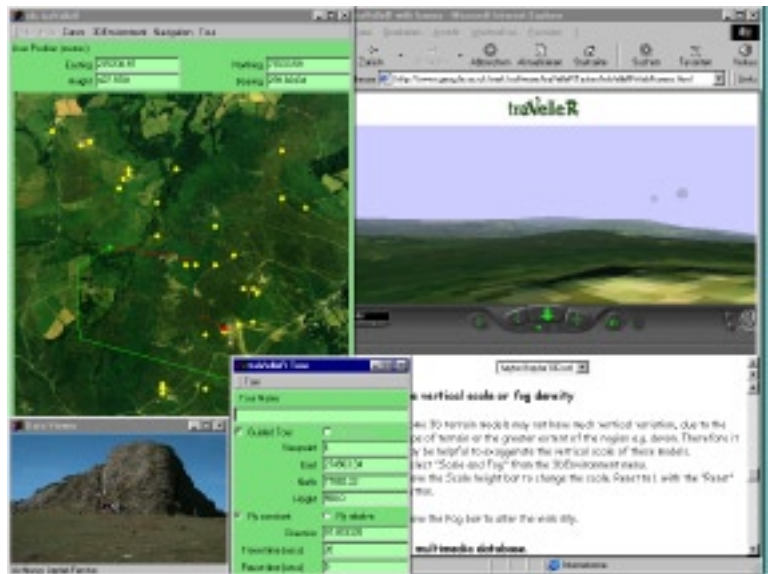


Abbildung 16: Screenshot vom traVelleR.

ren auswählen und einen Flug digitalisieren. Neben der Möglichkeit GPS-Daten zu importieren, können wir zu Hotspots andere Medien wie Webseiten, Bilder und Filme aufrufen.

Das USGS (U.S. Geological Survey) visualisiert geologische Messungen im Bereich von Vulkanen (<http://virtual.er.usgs.gov/>, Abb. 15). Neben der Möglichkeit verschiedene Ebenen zu- und abzuschalten, sehen wir Messungen in ihrem zeitlichen Verlauf.

Kaum zu finden sind Anwendungen, die abstraktere Daten wie z. B. Statistiken veranschaulichen. In besonderer Weise setzt J. SWANSON 1999, S. 186 die VRML-Technologie ein. In seiner *VRML Prism Map* (Abb. 17) ändern eine Reihe von Prismen ihre Höhe in einer Animation über mehrere Jahre. Jedes Prisma entspricht einer politischen Einheit. Berührt ein Betrachter ein Prisma, wird ihm der statistische Wert angezeigt. Zu bemängeln ist aber, dass die Höhe der Prismen der Anzahl eines Bewässerungssystem in diesem Gebiet entspricht. Wie im Zweidimensionalen sollte auch im 3D absolute Zahlen nicht auf Flächen dargestellt werden. Es ist selbstverständlich, dass dann auf grösseren Flächen auch höhere Prismen stehen.

Wir haben gesehen, dass die Einsatzmöglichkeiten von VRML in der Kartografie vielfältig sind und dass VRML schon in den verschiedensten Bereichen eingesetzt wird. Dennoch ist das Potential das in VRML steckt noch lange nicht ausgenutzt.



Abbildung 17: Screenshot von der VRML Prism Map.

6. Die Halbinsel Methana

Im diesem Abschnitt lernen wir die Halbinsel Methana und ihre Eignung für ein 3D Informationssystem kennen (Abb. 18 - 21).

6.1 Übersicht

Methana liegt ungefähr 70 km südwestlich von Athen.

„Obwohl Methana durch eine etwa 300 m breite Landbrücke, dem Isthmus Steno, mit der Peloponneshalbinsel verbunden ist, zählt die Vulkanhalbinsel geologisch zu den Saronischen Inseln und zum Ring der Kykladenvulkane. Der vulkanische Ursprung ist die Ursache für zahlreiche Thermalquellen. Sie machten die Bäder Methanas besonders bei den Griechen zu einem beliebten Ziel für einen sommerlichen Kururlaub. Die einzigartige Natur, die besonderen klimatischen Verhältnisse und die Heilquellen machten Methana schon seit vorgeschichtlicher Zeit zu einem beliebten Anziehungspunkt. Die Reste uralter Kulturen sind an vielen Stellen der Halbinsel zu entdecken. Besondere Berühmtheit erlangte Methana durch den geschichtlichen Vulkanausbruch beim heutigen Dorf Kaimeni Chora im Nordwesten der Halbinsel. Das gewaltige Ereignis wurde sogar von so berühmten antiken Schriftstellern wie Ovid, Pausanias und Strabon beschrieben. Lange vor den ersten Vulkanausbrüchen ragten die Karstgebirge Asprovouni und Krassopanagia aus dem Meer. ... Im Frühjahr wachsen auf

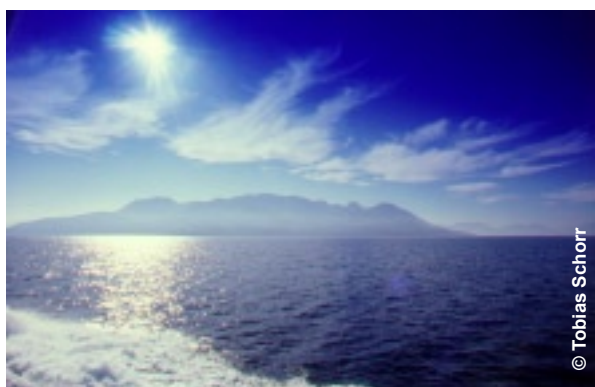


Abbildung 18: Methana von Ägina gesehen.

dem kalkigen Boden zahlreiche seltene Orchideen und Kräuter. Methana ist also ein ideales Ziel für Erholungssuchende, Naturliebhaber, Hobbyforscher und Wanderer!“ (T. SCHORR 1996, S. 3)

6.2 Eignung der Halbinsel Methana

Aus welchem Grund bietet es sich an, für eine kleine, unbekannte griechische Halbinsel ein 3D-Informationssystem zu entwickeln?



Abbildung 19: Antike Reste bei der Kapelle Agios Panteleimonas.

Für eine 3D-Visualisierung ist es von Vorteil, wenn es sich um ein abgeschlossenes Gebiet wie eine Insel handelt. Die Begrenzung des Modells bildet dann immer das Meer, und es gibt keine angeschnittenen Geländekanten, an denen das Modell abrupt aufhört. Da Methana eine Halbinsel ist, wird das Gelände im Süden an der Verbindung zur Peloponnes angeschnitten. Ansonsten bildet die Küstenlinie den Abschluss des Modells.

„Mit einer Ausdehnung von 9 mal 10 Kilometern und einer Fläche von 55 Quadratkilometern ...“ (L. HURNI 1995c, S. 10) handelt es sich um ein überschaubares Gebiet, so dass die Datenmenge relativ klein bleibt und sie noch in akzeptabler Zeit übers Internet zu übertragen ist. Für einen Prototypen wie Methana 3D empfiehlt sich ein kleines Bearbeitungsgebiet, weil dann der Schwerpunkt der Arbeit mehr auf der Entwicklung von Techniken als auf der Erzeugung und der Verarbeitung von grossen Datenmengen liegen kann.

Mit einer Reliefenergie von 740 Metern und durch die vielen Vulkankegel ist es reizvoll, Methana in einem 3D-Modell zu präsentieren. Für Methana 3D bekommt man deshalb ohne Überhöhung ein ansprechendes Bild.



Abbildung 20: Im Innern des Vulkans von Kameni Chora.

6.3 Quellen für Methana 3D

H. FORBES & C. MEE (Ed.)1997: A Rough and Rocky Place: Bericht über die archäologischen Untersuchungen der University of Liverpool von 1984-87 auf Methana, Beschreibung der physischen Geografie und der Geschichte der Besiedlung. Das Buch enthält eine ausführliche Liste der Fundorte, der Fundstätten und der Kirchen auf Methana, die sich gut in eine Datenbank integrieren liessen.

B. KAGIA 1995: Die Inseln des Saronischen Golfes: touristischer Führer des gesamten Saronischen Golfes. Er enthält ein kurzes Kapitel über Methana.

METHANA o. J.: touristische Multimedia-CD-ROM.

L. HURNI 1995c: Dissertation u. a. über Methana: In seiner Doktorarbeit hat L. Hurni am Beispiel von Methana untersucht, wie eine topografische (L. HURNI 1995c) bzw. eine geologische (L. HURNI 1995b) Karte vollständig auf digitalem Wege erstellt werden kann ohne Qualitätsverlust im Vergleich zur konventionellen, analogen Herstellung. Im Rahmen

der Dissertation ist ein digitales Höhenmodell entstanden, das in Maschen von 25 Metern die Form des Geländes beschreibt.

L. HURNI 1995b: geologische Karte von Methana mit lateinischer Beschriftung.

L. HURNI 1995a: topografische Karte von Methana: Von dieser Karte, die alle üblichen Elemente einer topografischen Karte enthält, sind alle Karteninhalte im Microstation DGN-Format als Vektoren vorhanden.

Diverse Manuskripte für Methana Reiseführer (z. B. T. SCHORR 1996) bzw. Begleitführer zur geologischen Karte in Deutsch, Englisch und Griechisch.

284 digitalisierte Fotografien im Bitmap-Format von Tobias Schorr, die er freundlicherweise für Methana 3D zur Verfügung gestellt hat.



Abbildung 21: Das Dorf Megalochori auf Methana.

Skizzen und mündliche Informationen von Tobias Schorr und Lorenz Hurni, die beide sehr gute Ortskenntnisse besitzen.

Durch diese umfangreichen Quellen konnte schon im vorhinein sichergestellt werden, dass Daten schnell zur Verfügung gestellt werden und dadurch keine unnötigen Wartezeiten entstehen.

7. Objekte auf Methana 3D

Welche Objekte soll Methana 3D beinhalten?

Das Grundgerüst der Insel bildet das digitale Geländemodell (DGM). Darauf sollen flächenhafte Inhalte wie Hypsometrie oder Bodenbedeckung dargestellt werden. Auf dieses DGM werden alle weiteren Objekte platziert. Zu den linienförmigen Objekten zählen alle Geometrien, die entlang einer Folge von Punkten positioniert werden. Natürlich ist die Darstellung trotzdem dreidimensional. Punktförmig werden all die Objekte genannt, bei denen eine individuelle oder wiederverwendbare, dreidimensionale „Signatur“ an einem Punkt platziert wird. Für diesen Punkt werden die drei Koordinaten und eventuell noch eine Orientierung benötigt.

Betrachten wir die einzelnen Objekte (Abb. 22).

7.1 Topografische Basiselemente

7.1.1 Punkthafte Objekte

In Methana 3D werden nur Kirchen und Kapellen dargestellt. Ursprünglich sollten alle Häuser einbezogen werden, ein Versuch zeigte jedoch, dass die Leistungsfähigkeit der heutigen Rechner noch nicht ausreicht, solche Datenmengen zu verarbeiten.

7.1.2 Linienhafte Objekte

Bäche und Strassen sind als linienhafte Objekte vorhanden. Bei den Strassen sind alle Hauptstrassen nach L. HURNI 1995a und die wichtigsten sonstigen Strassen eingetragen. Dazu sei angemerkt, dass neu ausgebaute Strassen nach den Angaben von T. Schorr ebenfalls Verwendung gefunden haben.

7.1.3 Flächenhafte Objekte

Flächenhafte Objekte wurden direkt in das DGM eingearbeitet oder als Rastertextur über die Insel gelegt. Zum einen kann die Hypsometrie, zum anderen kann die Bodenbedeckung betrachtet werden. Bei der Hypsometrie werden luftperspektivische Höhenstufen verwendet. Zwischen den Stufen wird die Farbe interpoliert. Die Bodenbedeckung geht streng genommen über den Inhalt der topografischen Basiselemente hinaus, wird aber trotzdem schon hier erwähnt. Berücksichtigt wurden Brachland, Wald, Gestrüpp und Kulturland. Bebautes Land kann ebenfalls betrachtet werden.

7.2 Thematische Elemente

7.2.1 Geologie

Die einzigen Elemente, die Informationen zur Geologie beinhalten, sind die Vulkane. Auf den wichtigsten Vulkanen wurden grüne Infotafeln positioniert, über die wir weitere Informationen abrufen können.

Es ist denkbar, das anstehende Gestein flächenhaft über das DGM zu projizieren; es wurde aber in dieser Version von Methana 3D noch nicht verwirklicht.



Abbildung 22: Verschiedene Objekte auf Methana 3D.

7.2.2 Archäologie

Wie bei der Geologie wurde die Archäologie nur mit Infotafeln berücksichtigt. Bei wichtigen historischen Fundstätten können wir uns über die braunen Infotafeln nähere Informationen anzeigen lassen. Die Fülle an Daten aus H. FORBES & C. MEE (Ed.) 1997 hätte den Rahmen dieser Arbeit gesprengt.

7.2.3 Tourismus

Da es sich um ein touristisches Informationssystem handelt, gehören die meisten thematischen Elemente zum Tourismus. Als linienhafte Objekte sind Wanderwege integriert. Diese besitzen einen Link, über den man sich über den Verlauf der Wanderung und deren Dauer informieren kann. An Gebäuden wurden die Klassen Hotel und Unterhaltung eingeführt. Unter Unterhaltung finden sich z. B. Cafés, Thermalbäder und Discotheken. Rote Infotafeln verweisen auf die Dörfer. Alle übrigen interessanten Sehenswürdigkeiten sind in den gelben Infotafeln als schöne Orte zusammengefasst.

8. Integration und Signaturierung der Objekte in Methana 3D

8.1 Das Grundgerüst, das DGM

8.1.1 Grundlagen

Ein digitales Höhenmodell (DGM) beschreibt gewöhnlich eine Oberfläche, meist die Erdoberfläche, durch eine Menge von Punkten. Die Genauigkeit des Modells hängt direkt von der Anzahl der Punkte ab. Es gibt zwei weitverbreitete Datenstrukturen zur Verwaltung von DGMs.

Bei einem *regelmässigen Gitternetz* (auch *Grid* genannt) sind die Höhen in einer zweidimensionalen Matrix gespeichert. Über die bekannten Koordinaten des Gitterursprungs und der festen Maschenweite können die Höhenpunkte im Raum positioniert werden. Der Vorteil dieser Datenstruktur sind einfache Datenverwaltung, einfacher Datenzugriff und einfache Rechenalgorithmen.

Dreiecksnetze bestehen aus ungleichmässig geformten Dreiecken (Abb. 23). Wir nennen sie im Weiteren *TIN* (Triangulated Irregular Network). Der grosse Vorteil eines TIN liegt darin, dass die Dichte der Stützpunkte davon abhängig gemacht werden kann, wie bewegt das Gelände ist. Eine gleichmässige Oberfläche lässt sich schon durch wenig Punkte in ausreichender Qualität beschreiben. In Gebieten mit bewegter Oberfläche verdichtet sich das Netz der Stützpunkte (M. ZANINI 1998, S. 24f).

8.1.2 Umsetzung

Als Grundlage für das DGM in Methana 3D diente ein Höhenmodell, das L. Hurni in seiner Dissertation angefertigt hat. Dieses Grid mit einer Maschenweite von 25 Metern hat er aus den Höhenkurven der topografischen Karte von Methana (L. HURNI 1995a) abgeleitet. Die ASCII-Datei wurde in Arc/Info eingelesen und dort als *LATTICE* (arc\lat25) gespeichert.

Wegen der kompakteren Datenspeicherung und dem Potential zu schnellen Berechnungen integrieren wir das DGM als TIN. Um eine weitere Beschleunigung der Darstellung zu erreichen, soll das Gelände je nach Entfernung vom Betrachter unterschiedlich aufgelöst sein (LOD). Also leiten wir aus dem *LATTICE* zwei

TINs ab. In Arc/Info kann der maximale Fehler angegeben werden, den ein Punkt im TIN von der Geländeoberfläche im *LATTICE* haben darf. Ein niedriger Fehler ergibt ein genaues (tin\tin25m7), ein hoher Fehler ein weniger genaues (tin\tin25m20) Modell (s. Listing 1.1: importDHM.aml).

Ein LOD-Knoten für das Gelände kann nur sinnvoll eingesetzt werden, wenn das Gelände in einzelne Kacheln zerlegt wird. R. ZEDI 1998, Kap. 6.2.4, weist darauf hin, dass in einem solchen Fall „zwei Teilstücke mit unterschiedlichem Detaillierungsgrad an den Schnittstellen nicht mehr exakt zusammen“ passen. Um das zu verhindern, werden alle Schnittpunkte der Kachelgrenzen mit dem hoch aufgelösten TIN gesucht. Von den gefundenen Punkten wird die Höhe berechnet (tin\clipnt). Nun wird sowohl das hoch aufgelöste als auch das niedrig aufgelöste TIN mit den neuen Punkten ergänzt (tin\mettin7, tin\mettin20, s. Listing 1.2: createtins.aml). Wird jetzt eine Kachel herausgeschnitten, bilden die neuen Punkte die Begrenzung der Kachel. Weil bei beiden

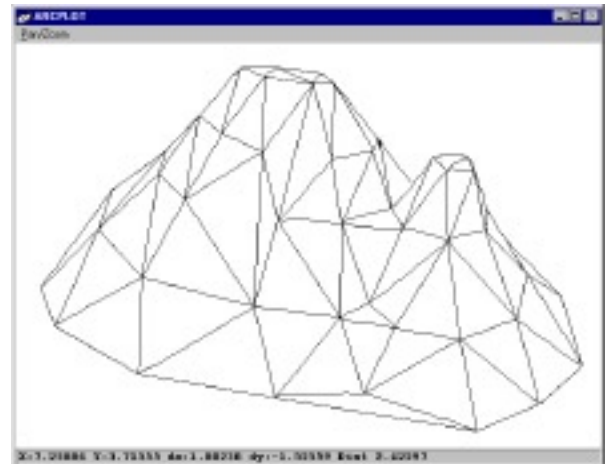


Abbildung 23: TIN (Triangulated Irregular Network).

Auflösungen die selben Randpunkte einbezogen sind, gibt es auch stetige Übergänge zwischen unterschiedlich aufgelösten Kacheln. Leider stimmt die Theorie nicht mit der Praxis überein. Vereinzelt treten immer noch Lücken auf, die vermutlich auf Rundungsfehler bei den Berechnungen zurückzuführen sind.

Nach der Berechnung der vollständigen DGMs, wird jede Kachel einzeln extrahiert, in eine ASCII-Datei exportiert und mit einem Perl-Skript in VRML umgewandelt (s. Listing 1.3: meshes.aml). Es werden pro Kachel drei VRML-Dateien erzeugt: das Netz in niedriger Auflösung (z. B. net10l.wrl), das Netz in hoher Auflösung (z. B. net10.wrl) und der LOD-Knoten (z. B. netlod10.wrl), der die Netze einschliesst.

Das Perl-Skript, das die VRML-Datei berechnet, betrachten wir ausführlicher.

Nach dem Header und allgemeinen Angaben für VRML werden die Koordinaten der Stützpunkte herausgeschrieben. Dabei ist zu beachten, dass die y- und die z-Koordinate vertauscht werden müssen, weil in VRML, wie in der Computergrafik üblich, die y-Koordinate vertikal auf dem Bildschirm ausgerichtet ist, also die Höhe beschreibt. Weiter muss die Richtung der z-Koordinate umgekehrt werden, sonst ist die Darstellung am Bildschirm spiegelverkehrt. Weil das DGM keine hohen Anforderungen an die Genauigkeit stellt, können die Koordinaten gerundet werden. Wird auf ganze Zahlen gerundet, kann die Grösse der VRML-Datei deutlich verkleinert werden.

Im zweiten Schritt werden die Höhenfarben berechnet. Bestimmte Höhen entsprechen vordefinierten Farbwerten. Punkte, die auf diesen Höhen zu liegen kommen, werden entsprechend eingefärbt. Die Farben aller dazwischen liegenden Punkte sind zu interpolieren. Weil dem DGM je nach Thema verschiedene Farben zugewiesen werden, empfiehlt es sich, die Farbinformationen in einer separaten Datei zu speichern.

Arc/Info speichert in der Exportdatei Knoten, Kanten und Dreiecke hintereinander. Im VRML *IndexedFaceSet*, das zum Speichern von TINs geeignet ist, werden die Dreiecke dagegen direkt aus den Knoten aufgebaut. Unser Perl-Skript muss also die Verweise der Dreiecke auf die Kanten in der ASCII-Datei direkt auf die Punkte zurückführen und für VRML speichern.

Im letzten Schritt wird der Abschluss der VRML-Datei ausgegeben (s. Listing 2.1: gen2vrml.pl).

8.2 Farbe in der Welt, verschiedene Oberflächenfarben

8.2.1 Grundlagen

Das DGM wird in VRML als *IndexedFaceSet*-Knoten gespeichert. Ein *IndexedFaceSet* können wir mit zwei Methoden einfärben. Im ersten Fall beeinflusst ein Attribut das gesamte Objekt. Wie jeder andere Geometrieknoten auch, beschreibt der *Appearance*-Teil des Objekts sein Aussehen. Die einfachste Möglichkeit, ein Objekt einzufärben, ist, dort eine Farbe anzugeben. An gleicher Stelle können wir auch eine

Textur definieren, die über die Oberfläche gerendert wird.

Die zweite Möglichkeit gibt jedem Stützpunkt (*ColorPerVertex*) oder auch jedem Polygon (*ColorPerFace*) eines *IndexedFaceSets* eine individuelle Farbe. Dazu wird im *color* Attribut eine Liste von Farben gesetzt.

8.2.2 Umsetzung

In Methana 3D sind drei Ansätze ausgeführt. Im ersten Beispiel wird jedem Stützpunkt des Geländes eine Farbe zugewiesen. Die Farbe zwischen den Punkten interpoliert das VRML Plug-in, um eine kontinuierliche Farbverteilung zu erreichen. Wie das zu erreichen ist, wenn die Farbe von der Höhe des Punktes abhängt, wurde im vorigen Abschnitt erklärt. Für andere Flächenfarben, wie z. B. die Bodenbedeckung, ist eine Polygondatei Voraussetzung, die Attribute der verschiedenen Ausprägungen enthält. Ähnlich dem Verfahren, das M. SUTER 1997, S. 51f, als *Rendring ohne texture mapping* beschreibt, untersucht man in welchem Polygon sich der Stützpunkt befindet (s. Listing 1.4: vege.aml) und weist ihm eine Farbe zu (s. Listing 2.2: attcolor.pl). Die Lage der Stützpunkte wird aus dem schon bestehenden VRML-Netz gelesen (s. Listing 2.3: net2point.pl). Dadurch sind sowohl Anzahl als auch Reihenfolge konsistent. Der Nachteil dieser Methode ist, dass es keine scharfe Begrenzung zwischen den Farbflächen gibt, z. B. gibt es keinen scharfen Waldrand.

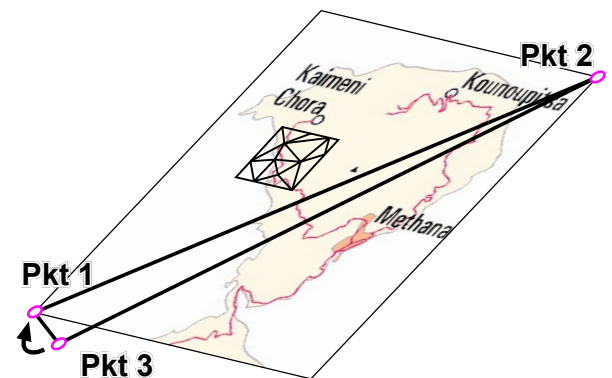


Abbildung 24: Textur wird auf die maximale Ausdehnung angepasst. Nur die Bereiche mit Polygonflächen werden dargestellt. Wird Pkt 3 mit Pkt 1 gleich gesetzt, ist die Fläche unsichtbar. Die Textur ist nur auf der Kachel sichtbar.

Der zweite Ansatz verbessert den Nachteil. Hier liefern Rasterbilder der Themen die Farbinformationen (Texture mapping). In Methana 3D besitzt jede DGM-Kachel ein Dreieck, das die Gesamtausdehnung von Methana beschreibt. Das Dreieck ist

unsichtbar, weil es nur aus den zwei Punkten einer Diagonalen besteht, also zwei der Dreieckspunkte identisch sind (Abb. 24). VRML positioniert eine Textur ohne Texturkoordinaten automatisch auf die maximale Ausdehnung des Objekts. Weil jede Kachel das Dreieck der Diagonalen besitzt, haben sie alle die selbe maximale Ausdehnung. Es gibt ein einheitliches Rasterbild, das sich nun ohne Versatz und unschönen Übergang über alle Kacheln legt. Aber auch hier gibt es Nachteile. Mit dem Cosmoplayer dauert das erste Rendern der Rasterbilder auf einen durchschnittlichen PC ungefähr zwei Minuten. Die weitere Bildwechselrate ist wieder schneller, verlangsamt sich aber merklich gegenüber dem Rendering ohne texture mapping. Ausserdem sieht ein Betrachter, wenn er zu nah am Gelände ist, immer die einzelnen Pixel. In Methana 3D gibt es nur für die Bodenbedeckung Texturen.

Der dritte Ansatz wurde bei den Siedlungsflächen angewendet. Das grafisch beste Ergebnis wird erzielt, wenn für jede Farbfläche ein eigener Ausschnitt des DGM mit einheitlicher Farbe erzeugt wird. Die Flächen werden aus dem gesamten DGM herausgestanzt (s. Listing 1.5: settlement.aml). Der Ablauf ist ähnlich der Erzeugung der LOD Kacheln (s. o.), nur dass die Clipping-Polygone durch einen Cursor automatisch durchlaufen werden. Vor dem ersten Polygon muss eine leere Datei erzeugt werden (s. Listing 2.4: newfile.pl). Jedes Polygon erzeugt dann einen selbstständigen LOD-Knoten für die Siedlungen (s. Listing 2.5: settlement.pl). Am Ende schliesst der Trailer die VRML-Datei ab (s. Listing 2.6: settlementend.pl). Der Übergang zur Nachbarfläche ist bei einem solchen Ansatz diskret und unabhängig von einer Pixelstruktur. Nachteilig wirkt sich aus, dass die Stützpunkte des DGM mehrfach abgespeichert werden und bei der Darstellung zusätzliche Polygone berechnet werden, obwohl ein Teil davon unsichtbar ist.

8.3 Linienhafte Objekte

8.3.1 Grundlagen/Allgemeines

Für alle linienhaften Objekte definieren wir einen Querschnitt und ziehen diesen über die Länge der Linie auf. In der Computergrafik wird das extrudieren genannt. Das Objekt soll immer genau auf der Geländeoberfläche liegen und nirgends versinken oder überstehen. Deshalb benötigen wir pro Dreiecksfläche des DGMs mindestens zwei Stützpunkte der Linie, jeweils an den Rändern der Fläche. Um Polygone zu sparen und die Bildwechselrate zu erhöhen, vermeiden wir alle Stützpunkte, die nicht auf

den Dreiecksrändern zu liegen kommen. Die Vorgehensweise besteht aus folgenden Schritten.

- Auswählen der Linien
- Löschen der Knoten, die genau zwei Segmente verbinden (bleiben als Stützpunkte erhalten)
- Schnittpunkte der Linien mit den Kanten des DGM suchen
- Höhen der Punkte auf dem DGM berechnen
- Aus den gefundenen Punkten ein VRML-Objekt generieren (Signaturierung)

Entfernt sich ein Betrachter über eine bestimmte Distanz hinaus, schalten sich die VRML-Objekte mit Hilfe eines LOD-Knotens ab. Das hat den Vorteil, dass die Bildwechselrate gesteigert wird und dass die Objekte nicht in das dann aktive, niedrig aufgelöste DGM versinken. Weil bei langen Objekten ein LOD nicht sinnvoll ist, werden alle Linien in kleine Stücke zerteilt, jeweils mit einem eigenen LOD.

Obwohl nach der Theorie alle Linien exakt auf dem Gelände liegen müssen, zeigte sich, dass es an wenigen Stellen zum Teil beträchtliche Abweichungen gibt, deren Ursache nicht nachvollzogen wurde.

8.3.2 Umsetzung

Beginnen wir mit den einfachsten Linienobjekten, den Bächen. Die Auswahl der gewünschten Flüsse geschieht interaktiv in Arc/Info. Daraus wird eine neue Coverage (Arc/Info-Vektordatei) erzeugt, die für die Weiterverarbeitung, wie oben beschrieben, verwendet wird (s. Listing 1.6: river.aml). Auf die Umwandlung von der Mittellinie zum VRML-Bach gehen wir genauer ein (s. Listing 2.7: river.pl).

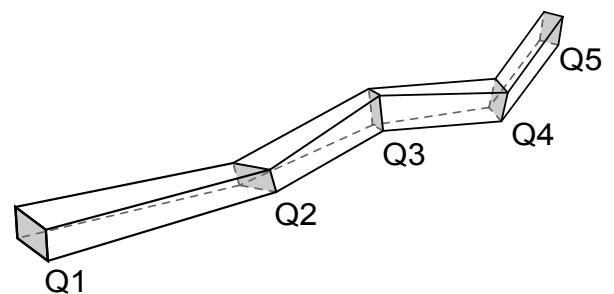


Abbildung 25: In einem VRML Extrusion Node sind die Querschnitte nicht immer gleich ausgerichtet.

Die Bachsignatur erhält in Methana 3D einen Kreis als Querschnitt. In VRML haben Extrusion-Knoten die Eigenschaft, dass sich die Orientierung des Querschnitts je nach Lage der Stützpunkte verdreht (Abb. 25). Ein nicht punktsymmetrischer Quer-

schnitt muss dann recht aufwendig in die richtige Lage gedreht werden. Wegen der Punktsymmetrie des Kreises können die Bäche problemlos als Extrusion-Knoten gespeichert werden. Am Anfang jedes Segments wird der Kreisquerschnitt durch ein 16-Eck angenähert. Dann müssen nur noch die Stützpunkte der Linien in richtiger Reihenfolge angegeben werden.

Der Ablauf zur Erzeugung der Wanderwege entspricht dem der Bäche, nur dass die Signatur eine andere Farbe bekommt und zusätzlich ein Link verankert wird, der auf Informationsseiten verweist (s. Listing 1.7: hike.aml, Listing 2.8: hike.pl).

Deutlich komplexer sind die Berechnungen für die Strassen. Im Gegensatz zu den zwei vorherigen Objektarten werden Strassen automatisch nach der Strassenklasse ausgewählt. Der weitere Ablauf entspricht dem der bisherigen Beispiele. Weil der Querschnitt der Strassensignatur nicht punktsymmetrisch ist, kann keine Extrusion-Knoten in VRML verwendet werden. Der Querschnitt besteht aus drei Linien, nämlich einer waagrechten Linie für den Strassenbelag und zwei senkrechten Linien links und rechts für Stützmauern (Abb. 26). Die Stützmauern verhindern die Sicht unter die Strasse und geben ihr eine akzeptable Begrenzung. Beide Mauern reichen bis

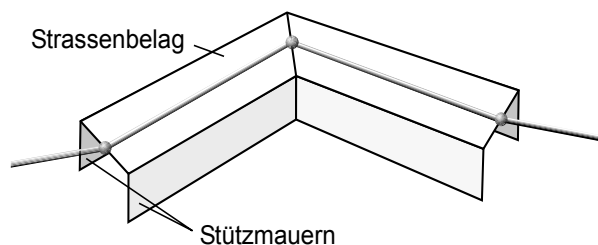


Abbildung 26: Aufbau der Strassen.

zum Meeresboden. Damit gibt es auch bei steilem Gelände keine Lücken zwischen Mauer und Gelände. Weil die Strasse leicht erhöht über dem Gelände verläuft und von der Mittellinie nach links und rechts ausgedehnt wird, ist die Fahrbahnbreite von der Neigung des Geländes abhängig, was aber kaum störend auffällt. Die vier Punkte pro Querschnitt werden einzeln berechnet und in einem IndexedFaceSet-Knoten gespeichert (s. Listing 1.8: roads.aml, Listing 2.9: roads.pl). Wer den Algorithmus nachvollziehen möchte, sei auf den Programmcode in Listing 2.9 verwiesen.

8.4 Punkthafte Objekte

8.4.1 Allgemeines

Punktobjekte können in zwei Gruppen eingeteilt werden. Zur ersten Gruppe gehören Objekte, die in ihrer Form und Ausprägung individuell gestaltet sind. Wir müssen ihre Geometrie jedesmal gesondert definieren, wenn sie vorkommen. Bei der anderen Gruppe kann für mehrere Objekte jeweils die selbe Geometrie verwendet werden. Eine solche Geometrie, die stets eine Vereinfachung der Wirklichkeit ist, nennen wir *Signatur*. Die Signatur wird manchmal leicht modifiziert, z. B. kann die Farbe geändert werden.

Auf Methana 3D gibt es keine individuellen Objekte. Wir finden nur die zwei Hauptklassen Gebäude und Informationstafeln, die in Unterklassen wie z. B. Hotel, Kirche oder Kapelle aufgeteilt sind. In VRML sind die Signaturen durch *Prototypen* definiert. Der Prototyp *Infotafel* hat z. B. die Attribute Position und Farbe. Wird eine Infotafel platziert, müssen nur die richtige Position und eine Farbe angegeben werden. Die Geometrie ist schon im Prototypen definiert.

Für alle punktförmigen Objekte kann der selbe Ablauf angewandt werden:

- Auswählen der Objekte
- Finden der Punkte, die ihre Lage repräsentieren
- Berechnen der Höhen der Punkte auf dem DGM
- Zuweisen der 3D-Koordinaten an den VRML-Prototypen

Weil beim endgültigen Aufbau von Methana 3D alle punktförmigen Objekte in einer Datenbank gespeichert werden (s. u.), sind die Ergebnisse der Berechnungen entweder SQL-Befehlsdateien oder Koordinatenlisten, die von Hand zu SQL-Befehlsdateien editiert werden.

8.4.2 Umsetzung

Informationstafel: Die Position der Informationstafel musste neu digitalisiert werden (s. Listing 1.9: positionMap.aml, arc\metsights), weil es keine digitalen Ausgangsdaten gab. Die Auswahl hat sich in erster Linie danach gerichtet, ob Fotografien vorhanden waren und die Tafeln mit weiterer Information verlinkt werden konnten. Die Signatur der Infotafeln besteht aus zwei LOD-Stufen. Von weitem sieht der Betrachter eine farbige Kugel auf einer Stange.

Nähert er sich einer Tafel, wird die Kugel durch ein Schild mit Text ersetzt, das zusammen mit der Farbe auf die Thematik des Objekts hinweist. Die Tafeln sind mit Webseiten verlinkt, die das Objekt mit Text und Bild erklären. An Unterklassen sind Dörfer (rot), Vulkane (grün), historische Plätze (braun) und sonstige schöne Plätze (gelb) unterschieden.

Hotel, Unterhaltung: Die Hotels und Unterhaltungsbauwerke sind ebenfalls neu digitalisiert (arc\methotel).

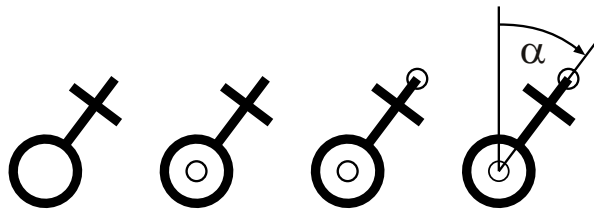


Abbildung 27: Berechnung der Position und Ausrichtung einer Kirche.

Es muss eine Coverage erzeugt werden, die nur die gewünschten Häuser enthält. Die Häuser dürfen nur aus vier Seiten bestehen, sonst werden sie ignoriert. Die Lage des Mittelpunkts ist das arithmetische Mittel der x- bzw. y-Koordinaten. Die Ausrichtung berechnet sich aus der längsten Seite. (s. Listing 1.10: hotel.aml, Listing 2.10: houses2pnts.pl, Listing 2.11: houses2sql.pl)

Kirche, Kapelle: Kirchen und Kapellen sind in der topografischen Karte von Methana (L. HURNI 1995a) vorhanden (arc\methaus). Die Signatur besteht aus einem Kreis und zwei sich kreuzenden Linien. Mit dem Mittelpunkt des Kreises wird die Position definiert. Die Ausrichtung α ergibt sich aus der Linie zwischen Mittelpunkt und dem am weitesten entfernten Punkt (Abb. 27). (s. Listing 1.11: churches.aml, Listing 1.12: chapels.aml, Listing 2.12: churches2pnts.pl, Listing 2.13: churches2sql.pl)

9. Datenbank für Methana 3D

9.1 Einführung

„Eine Datenbank ist eine selbständige, auf Dauer und für flexiblen und sicheren Gebrauch ausgelegte Datenorganisation, umfassend einen Datenbestand (Datenbasis) und die dazugehörige Datenverwaltung.“ (C. A. ZEHNDER 1987, S. 10)

Bei einer Datenbank (DB) sind die Daten streng vom Benutzer (Anwendung) getrennt. Die Daten werden zentral gespeichert und verwaltet. Jeder Benutzer bekommt nur Zugang zu den Daten, die er benötigt. Durch die Datenunabhängigkeit sind die Anwendungen nicht von Reorganisation und Erweiterung des DB-Systems betroffen, verfügen aber dennoch über die aktuellsten Daten. Auf die Daten kann gleichzeitig von mehreren Anwendungen zugegriffen werden. Weiter können sie wiederverwendet werden, bzw. neue Anwendungen können sie verwenden.

Der Datenbestand hat einen strukturierten Aufbau, so dass der Zugriff auf Einzeldaten schnell möglich ist. Mehrfachspeicherung kann vermieden werden.

Mit den heutigen Internettechnologien ist es kein Problem, von entfernten Rechnern auf eine zentrale DB, die auf dem sogenannten DB-Server liegt, zuzugreifen. Es wird dadurch möglich, von überall auf der Welt auf die selben Daten zuzugreifen.

Inhalte von Methana 3D sollen aus einem DB-Server abgerufen werden. Die Vorteile dieses Konzepts konkret für das Projekt sind folgende: Daten können einfach aktualisiert werden, ohne VRML-Dateien editieren zu müssen. Weil beim Aufbau von Webseiten direkt auf die Datenbank (DB) zugegriffen wird, sind sofort alle Änderungen verfügbar.

Die DB kann von autorisierten Benutzern online aktualisiert werden. Wenn z. B. ein neues Hotel eröffnet wird oder sich der Link zum Hotel ändert, kann die zuständige Stelle die Einträge selbst vornehmen, auch wenn die DB nicht vor Ort liegt.

Eine günstig aufgebaute DB kann gleichzeitig für mehrere Zwecke verwendet werden. Aus den Tabellen für Methana 3D werden einerseits VRML-Objekte erzeugt und andererseits HTML-Seiten mit Bildern und Texten.

In unserem Fall sind in der DB nur Punktobjekte gespeichert, d. h. Gebäude und Infotafeln. Prinzipiell ist es denkbar, auch Linien oder Flächen mit einer DB zu verwalten. Das Bereitstellen der VRML-Objekte würde dann länger dauern. Ausserdem ändern sich Objekte wie Strassen oder Bäche nicht so häufig wie die Attribute eines Gebäudes.

9.2 Aufbau der DB

Methana 3D benötigt zwei verschiedene Tabellen (s. Listing 5.1). In der Tabelle *sights* sind alle Sehenswürdigkeiten, in der Tabelle *buildings* alle Gebäude gespeichert.

Beiden Tabellen gemeinsam sind folgende Spalten:

id: Identifizierungsnummer, für jedes Objekt verschieden, allerdings haben Zeilen, die das selbe Objekt betreffen die gleiche ID

feature: Unterklasse des Objekts, damit wird das Erscheinungsbild der Signatur angepasst

name: Name des Objekts

target: Zielfenster oder Rahmen, in dem die Informationsseite mit Text und Bild aufgebaut werden soll

imageUrl: Pfad, in dem eine Fotografie des Objekts abgelegt ist

description: Text, der im Informationsfenster verwendet wird

externLink: Link, der auf externe Seiten verweist

xpos / ypos / zpos: Mittelpunkt des Objekts auf der Geländeoberfläche, sind mehrere Einträge für das selbe Objekt vorhanden und haben die selbe ID, dann darf nur ein Eintrag Koordinaten enthalten, alle anderen müssen auf ,0' gesetzt sein.

Spezifisch für die Tabelle *sights* sind diese Spalten:

campos: Kameraposition zum Betrachten des Objekts, bei mehrfachen IDs müssen sie bis auf einen Eintrag leer sein

camorient: Orientierung der Kamera, bei mehrfachen IDs müssen sie bis auf einen Eintrag leer sein

In *buildings* sind diese Angaben nicht nötig, weil sie dort direkt aus der Position des Objekts berechnet werden können. Bei *sights* ist die Kameraposition nicht berechenbar, weil sie immer so ausgerichtet ist, dass sie ungefähr der der Fotografie entspricht.

Nur in der Tabelle *buildings* kommen diese Spalten vor:

- orient*: Orientierung des Gebäudes
- size*: Grösse des Gebäudes
- color*: Farbe des Gebäudes

9.3 Objekterzeugung aus der DB

Am Beispiel der Infotafeln wollen wir erklären, wie Objekte aus der DB in Methana 3D eingebaut werden.

Der Betrachter klickt dazu auf Checkbox *Information signboards*. Ein mit der Checkbox verbundenes Javascript ruft über CGI ein Perl-Skript auf, das aus der DB alle Einträge der Tabelle *sights* abrufen. Perl erstellt nun eine VRML-Datei. Zu jedem Eintrag werden dazu die Eigenschaften Farbe und Text entsprechend der Spalte *feature* und die Position einem Prototypen *InfoSign* zugewiesen. Die dynamisch erzeugte Datei mit allen Infotafeln wird an den Client übertragen und vom VRML-Plug-in in die Welt eingebaut. Die Infotafeln können mit der Maus angeklickt werden, um Informationen über die Lokalität zu erhalten (Abb. 29).

9.4 HTML-Erzeugung aus der DB

Wie die Objekte werden auch die Informationsseiten, die mit den Objekten verlinkt sind, aus der DB erstellt. Wird eine Infotafel angeklickt, wird wieder ein CGI-Skript aufgerufen. Wie bei der Objekterzeugung stellt es eine Anfrage an die Datenbank. Diesmal wird nur das Objekt ausgegeben, das die ID der angeklickten Tafel hat. Aus der Zeile werden nun Informationen wie Name, Pfad des Bildes und der beschreibende Text zu einer HTML-Seite zusammengestellt. Sie wird dann wieder an den Client übertragen und im geeigneten Webbrowserfenster dargestellt. (Abb. 28)



Abbildung 29: Infotafeln, die aus der Datenbank erzeugt werden.

9.5 HTML-Listen aus der DB

Ein dritter Bereich wird dynamisch aus der DB erzeugt. Für ein Informationssystem ist es wichtig, nach Informationen suchen zu können. In der aktuellen Version von Methana 3D ist das auf recht einfache Weise gelöst. Zu verschiedenen Themen können wir uns alle Objekte auflisten lassen. Mit der Auswahl eines Elements werden wir direkt zum ausgewählten Objekt hingeführt.

Folgendes Beispiel soll uns den Ablauf einer Suche verdeutlichen. Ein Besucher möchte wissen, wo sich das Hotel Saronis befindet. Er lässt sich nun die Liste der Hotels zeigen. Dazu sucht ein CGI-Skript in der DB alle Hotels und stellt sie in einer HTML Auswahlliste (select-Tag) zusammen. Markiert der Suchende das Hotel Saronis wird er in der Welt so platziert, dass das Hotel genau vor ihm liegt. Die nötige Betrachterposition wird automatisch aus der Position des Hotels berechnet. Ausserdem wird noch die HTML-Informationssseite aus der DB generiert, wie oben schon beschrieben. Hier kann der Benutzer die Fotografie des Hotels und weitere Infos erhalten, z. B. einen Link zur Zimmerreservierung.

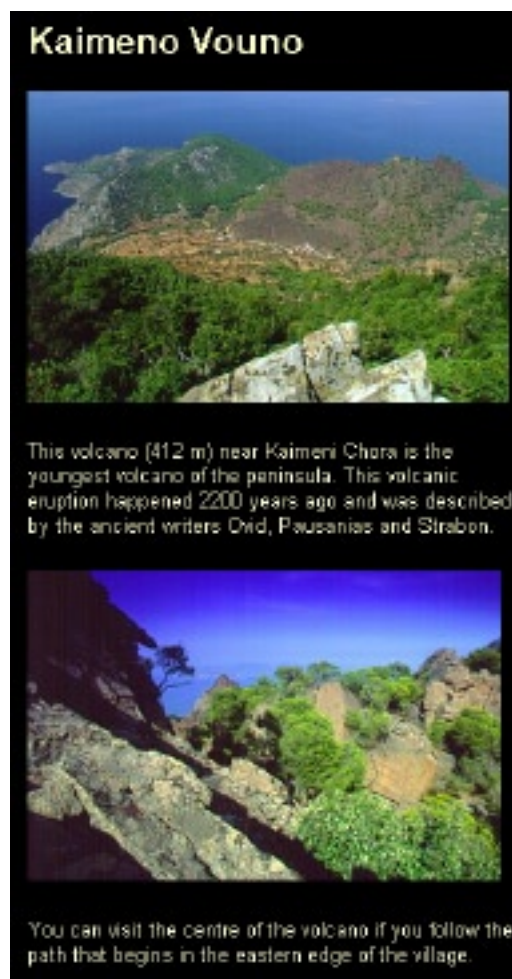


Abbildung 28: Aus der Datenbank erzeugte HTML Seite.

10. Funktionalität von Methana 3D

10.1 Übersicht

Besuchen wir Methana 3D, gelangen wir zu Beginn auf die Eingangsseite, die uns über die System-

voraussetzungen informiert. ‚Intelligente Listen‘ wählen, soweit es möglich ist, den benutzten Internet-Browser und das VRML-Plug-in aus. Damit kann uns aus einer Auswahl von Welten immer die angeboten werden, die für unser System am geeignetsten ist. Drücken wir jetzt auf START, öffnet sich das eigentliche Hauptfenster (Abb. 30).



Abbildung 30: Hauptfenster von Methana 3D mit Hauptmenü, VRML Welt, Übersichtskarte, Auswahlliste, Inforahmen und Legende (von links nach rechts, von oben nach unten).

Die Benutzeroberfläche von Methana 3D hat sechs Bereiche. Das Fenster des Internet-Browsers wird links vom grossen Rahmen mit der VRML-Welt beherrscht, in dem durch die Welt navigiert wird. Darüber schliessen der Titel und das Hauptmenü den Inhalt des Fenster ab. Mit dem Menü gelangen wir auf die verschiedenen Informationsebenen, auf denen wir weitere Aktionen ausführen können. Rechts vom VRML-Fenster ist im oberen Bereich eine Übersichtskarte platziert, die uns zeigt, wo auf der Halbinsel wir uns gerade befinden. Ausserdem können wir mit ihr schnell auf eine andere Stelle in der VRML-Welt springen. Rechts davon ist ein Bereich, der für Auswahllisten reserviert ist. Erst

wenn Listen angefordert werden, wird der Bereich gefüllt.

Im rechten unteren Teil des Fensters befindet sich der Bereich, in dem die Informationsebenen geladen werden (Inforahmen). Beim ersten Laden von Methana 3D ist die Steuereinheit zur Auswahl der Themen, die in der Welt vorhanden sein sollen, sichtbar. Wird die Informationsseite zu einem Objekt angefordert, so wird sie ebenfalls in diesen Bereich abgebildet. Zu guter Letzt ist links unten ein Rahmen, in dem Legenden und Copyright zu finden sind.

In den nächsten Abschnitten werden wir kennen lernen, wie die Funktionalität der einzelnen Bereiche umgesetzt wurde.

10.2 VRML-Welt

In diesem Bereich wird die VRML-Welt geladen und initialisiert. Die Funktionalität des Rahmens wird durch das VRML-Plug-in zur Verfügung gestellt. Als wichtigste Funktion übernimmt es die Darstellung der Geometrie. Je nach Plug-in können verschiedene Methoden der 3D-Computergrafik zum Rendering ausgewählt werden. Über ein Control-Panel (Abb. 31) ist es dem Besucher möglich, die Kamera zu beeinflussen. Die erste Methode bewegt die Position



Abbildung 31: Control-Panel von blaxxun Contact.

des Betrachters. Er kann fliegen, gehen, die Höhe ändern und den Kopf heben oder neigen. Gewöhnlich kann auch auf einen Untersuchungsmodus umgestellt werden, in dem die Objekte (natürlich nur im Gesamten) zu drehen, zu verschieben oder zu zoomen sind. Dem interessierten Benutzer eines VRML-Plug-ins sei geraten, mit der rechten Maustaste in die Welt zu klicken oder die Hilfefunktion auf dem Control-Panel zu suchen. An dieser Stelle wird nochmals darauf hingewiesen, dass das Plug-in das External Authering Interface (EAI) unterstützen muss, um die volle Funktion von Methana 3D ausführen zu können (s. Listing 3.1: metWorld.html).

Die Hauptdatei, aus der die VRML-Welt gelesen wird, ruft alle untergeordneten VRML-Dateien auf und übernimmt deren Steuerung (s. Listing 7.1: methana.wrl).

10.3 Hauptmenü

Das Hauptmenü erstreckt sich am oberen Fensterand über die gesamte Breite. Neben dem Titel befinden sich Links, die auf die verschiedenen Informationsebenen verweisen. Die mit einem Link versehenen Schriftzüge ändern ihre Farbe, wenn wir mit der

Maus darüber fahren. Dadurch werden wir darauf aufmerksam gemacht, dass hinter diesen Texten weitere Aktionen verborgen sind. Wird ein Text angeklickt, baut sich die gewünschte Seite im Inforahmen auf. Der Schriftzug bleibt aktiv, damit immer nachvollzogen werden kann, welche Seite gerade sichtbar ist. Sehr wichtig für alle Javascripts ist, dass sich im Arbeitsbereich dieses Rahmens alle Variablen befinden, die in Methana 3D benötigt werden (s. D. FLANAGAN 1998, S. 60, 208). (s. Listing 3.2: menu.html)

10.4 Übersichtskarte

Die Übersichtskarte, in dem die Position und die Ausrichtung des Besuchers durch ein kleines, rotes Dreieck zu sehen ist, ist unabdingbar, damit wir uns auf der Halbinsel zurechtfinden und nicht die Orientierung verlieren. Technisch umgesetzt wurde die Karte durch eine zweite VRML-Welt, die eine ebene Fläche enthält, auf die ein Rasterbild gelegt ist. Deshalb ist die Qualität nicht so hoch, wie sie mit einem normalen Rasterbild zu erreichen wäre. Dafür ist die Steuerung des Dreiecks überraschend einfach. In regelmäßigen Abständen wird die Gesamtorientierung der Kamera in der eigentlichen VRML-Welt abgerufen. Die erhaltenen Koordinaten werden direkt auf das Dreieck übertragen. Weil die Koordinaten der Karte exakt dem der VRML-Welt entsprechen, muss nicht einmal eine Transformation vorgenommen werden (s. Listing 4.1: init.js#Uebersichtskarte u. Listing 4.2: script.js#aktuelle Position).

Des Weiteren dient die Karte zur schnellen Positionierung an einem beliebigen Punkt auf Methana. Vor allem auf langsamen Rechnern ist man froh, wenn auf diese Art schnell navigiert werden kann. Mit einem ersten Klick legen wir den Ort fest. Bei jeder weiteren Bewegung der Maus wird jetzt die Kamera entsprechend ausgerichtet. Erst beim zweiten Klick bleibt die Orientierung erhalten. Wir können jetzt wieder wie gewohnt im Hauptrahmen navigieren. Damit wir uns nie unterhalb der Geländeoberfläche platzieren, wurde ein stark vereinfachtes Höhenmodell über die Karte gelegt. Die Koordinaten werden von diesem durchsichtigen DGM abgetastet. An der Küste werden wir auf eine niedrige Höhen gesetzt, im Inland auf eine hohe (s. Listing 4.1: init.js#Uebersichtskarte u. Listing 4.2: script.js#Kamera positionieren).

Leider hat sich gezeigt, dass das Positionieren mit zwei Mausklicken nicht intuitiv ist und viele Betrachter Mühe haben, sich richtig zu platzieren.

10.5 Auswahllisten

Eine Auswahlliste erhalten wir, wenn wir im Hauptmenü *select by list* wählen. Jetzt erscheint im Inforahmen eine Reihe von Tasten, die uns die gesuchte Liste öffnen (s. Listing 3.3: select.html).

Es gibt drei Bereiche von Listen. Im ersten sind allgemeine Kamerapositionen (Viewpoint) definiert, die uns zur Übersicht auf Methana, auf eine Ansicht von oben und andere Positionen führen. Dazu wird der ausgewählte Viewpoint im der VRML Welt aktiviert. Gleichzeitig wird der Viewpoint in einem Cookie (s. D. FLANAGAN 1998, S. 334 – 342) gespeichert. Betreten wir das nächste Mal Methana 3D befinden wir uns direkt an diesem Viewpoint. (s. Listing 3.4: viewpoints.html)

Der zweite Bereich führt uns zu den Sehenswürdigkeiten. Es sind vier thematisch verschiedene Listen abrufbar. Die Listen werden direkt aus der Datenbank generiert. Dazu existiert ein Skript, das je nach Parameter Ausschnitte aus der Tabelle in einer Liste zusammenstellt (s. Listing 6.1: listsights.pl, Listing 3.5: listvolcano.html). Wählen wir einen Eintrag aus der Liste aus, bekommen wir zum einen die Informationsseite mit einer Fotografie, zum anderen wird unsere Kamera in der VRML-Welt entsprechend der Fotografie ausgerichtet. Alle dazu nötigen Informationen wie Textinformationen oder Position der Kamera, werden direkt aus der DB abgerufen (s. Listing 4.2: script.js#Objekt Auswählen, Listing 6.2: createSide.pl, Listing 6.3: sightsvp.pl).

Mit dem dritten und letzten Bereich können wir Gebäude auswählen. Die Funktionsweise entspricht der vorigen (s. Listing 6.4: listbuidings.pl), nur dass die Kameraparameter nicht aus der DB abgerufen werden, sondern direkt aus der Position der Gebäude berechnet werden. Um die Gefahr zu verringern, dass sich der Betrachter plötzlich unterhalb des Geländes befindet, wird die Kamera immer zum Mittelpunkt der Insel ausgerichtet (s. Listing 6.5: buildingsvp.pl).

10.6 Inforahmen

In diesem Rahmen finden wir zum einen die Webseiten, die aus dem Hauptmenü geöffnet werden, und zum anderen die Informationsseiten, die aufgerufen werden, wenn wir auf ein aktives Objekt in der

VRML-Welt (Gebäude, Wanderwege...) klicken. Im Folgenden betrachten wir die einzelnen Seiten und lernen ihre Funktionalität kennen.

10.6.1 choose topic

Beim ersten Start von Methana 3D sehen wir im Rahmen das Auswahlmenü, mit dem die Themen, die in der VRML-Welt dargestellt werden, ein- und ausgeschaltet werden können (s. Listing 3.6: topic.html). Haken wir ein Feld ab, so wird das Thema dargestellt. Einige Objekte werden beim Starten von Methana 3D schon mit geladen, sind aber noch nicht sichtbar. Verwendet wird dazu ein VRML-Switch-Knoten. Das ist ein Gruppenknoten, der mehrere Kinderknoten besitzen kann, von denen aber immer nur einer oder keiner dargestellt wird. Ein einfaches Javascript, das durch das Anklicken einer Checkbox ausgelöst wird, sendet der VRML-Welt die Information, welcher Kinderknoten sichtbar sein soll (s. Listing 4.2: script.js#Funktionen interaktive Steuerung der Welt). Der grosse Nachteil bei diesem Verfahren ist, dass beim ersten Aufbau der Welt Objekte geladen werden, die gar nicht sichtbar sind. Das kostet unnötige Ladezeit. Verwendet wird es deshalb nur bei kleinen oder oft eingeschalteten Objekten, wie Strassen, Flüssen und Wanderwegen.

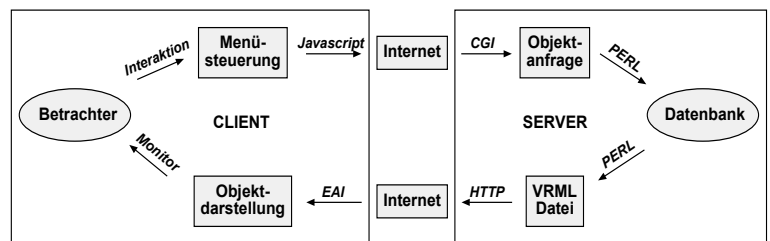


Abbildung 32: Vereinfachter Ablauf der Objektanfrage aus der Datenbank.

Die anderen Themen werden erst geladen, wenn wir ein Kästchen ankreuzen. In VRML sind dafür eigene, leere Gruppenknoten reserviert. Weiter wird ein VRML-Skriptknoten benötigt, der den Inhalt einer Url in den Gruppenknoten einbaut. Das Javascript sendet die Url an den Skriptknoten (s. Listing 4.2: script.js#Funktionen interaktive Steuerung der Welt). Wird eine Objektgruppe wieder ausgeschaltet, wird der gesamte Gruppenknoten geleert. Der Nachteil der zweiten Methode besteht darin, dass die Objekte jedesmal neu geladen werden müssen, wenn sie angeschaltet werden. Da heisst, dass erstens das Netz unnötig belastet wird und dass wir zweitens die ganze Zeit mit dem Internet verbunden sein müssen. Vorteilhaft ist, dass die Dateien erst geladen werden, wenn sie dargestellt werden sollen. Ausserdem ist das

die beste Möglichkeit in Kombination mit Abfragen aus einer DB, weil die beim Aufruf aktuellen Einträge Verwendung finden. Einen vereinfachten Ablauf der Objektanfrage aus der Datenbank zeigt die Grafik in Abb. 32.

Mit einer Auswahlliste wählen wir das Thema, mit dem das Gelände der Halbinsel eingefärbt wird. Dazu wird der Color-Knoten des Geländes ausgetauscht. Soweit vorhanden, können wir die Einfärbung auch mit Rastertexturen vornehmen, was aber lange Ladezeiten zur Folge hat. Auf diese Weise könnte auch ein Luft- oder Satellitenbild über das Gelände gelegt werden. Wegen den hohen Anschaffungskosten konnte leider keines zur Verfügung gestellt werden. Mehrere Prototypen übernehmen die komplizierte Steuerung, das Thema über alle Kacheln und LODs auszutauschen.

Zu beachten sind auf dieser Seite auch die Tasten, mit denen die Legenden zu verschiedenen Themen angezeigt werden.

10.6.2 select by list

Die Seite wurde schon bei den Auswahllisten beschrieben.

10.6.3 help / refresh

Diese Seite ist dafür reserviert, Erklärungen zur Bedienung von Methana 3D zu präsentieren. Ausserdem finden wir eine Übersicht über alle Legenden. Weil manche Web-Browser-Plug-in-Kombinationen Pro-

bleme haben, alle Einstellungen über das EAI nach VRML zu übernehmen, während die Welt lädt, ist hier die Möglichkeit gegeben, die Einstellungen zu aktualisieren.

10.6.4 links

Für alle die es nicht auf Methana 3D hält, ist hier eine Liste interessanter Links zusammengestellt, die weitere Informationen zu Methana bieten. Stand: September 1999.

10.6.5 about project

Diese Seite erklärt kurz, wie das Projekt Methana 3D zustande gekommen ist und welche Personen daran beteiligt sind.

10.7 Legende

In diesem Rahmen unter der VRML-Welt sehen wir die Legenden zu verschiedenen Themen. Legenden können wir unter den Menüs ‚choose topic‘ oder ‚help / refresh‘ aufrufen. Ausserdem befinden sich in diesem Rahmen die Angaben zum Copyright.

11 Schlussanalyse

11.1 Problembetrachtung

In diesem Abschnitt wollen wir Probleme betrachten, die bei der Erzeugung oder während der Laufzeit aufgetreten sind und nicht oder nur provisorisch gelöst wurden.

11.1.1 Probleme bei der Objekterzeugung

Obwohl die Linienobjekte und die Siedlungsflächen nach dem verwendeten Algorithmus genau auf dem DGM liegen sollten, muss festgestellt werden, dass dies nicht immer der Fall ist. Es gibt einige Stellen an denen ein Stützpunkt der Strasse unter dem Gelände verschwindet oder an dem der Bachlauf deutlich über dem Gelände liegt. Weil alle Objekte automatisch erzeugt werden, ist es recht schwierig diese Fehler in den Dateien zu identifizieren und eventuell per Hand zu korrigieren. Ein ähnliches Problem tritt bei den Siedlungsflächen auf. Es gibt Stützpunkte die zu hoch oder zu tief liegen, obwohl sie aus dem gleichen TIN herausgeschnitten wurden wie das normale Gelände.

Ein anderes Problem ist, dass manche Betrachter nicht erkennen, um was für Objekte es sich handelt ohne eine Legende zu sehen. Vielleicht hätten noch offensichtlichere Signaturen gefunden werden können. Nicht für alle Objekte ist eine Legende vorhanden.

11.1.2 Probleme mit klaren Lösungen

In diesen Bereich fallen die Probleme, für die die Lösung offensichtlich ist, aber nicht genügend Zeit war, um sie umzusetzen. Ausserdem stehen hier ungünstig verwirklichte Implementierungen, die mit einem gewissen Zeitaufwand besser programmiert worden wären.

Die erste Vereinfachung fällt uns auf, wenn wir Methana 3D mit einem unüblichen Plug-in starten. Hier erscheint immer nur die Nachricht, dass man die Anwendung auf Wunsch auch mit dem nicht vorgesehenen Plug-in starten kann. Die ursprüngliche Idee war, dass für verschiedene Plug-in – Browser-Kombinationen die passende Welten angeboten werden können. Leider ist bisher nur die Version für CosmoPlayer – Netscape oder Cortona – Netscape lauffähig.

Leider ist in Methana 3D keine vernünftige Hilfe Funktion integriert. Es sollte dem Betrachter möglich sein, alle Funktionen, die ausführbar sind, nachzulesen. Sonst wird er sie nicht finden. Dazu ist es auch wichtig, die richtigen Startseiten beim Öffnen festzulegen, damit der Besucher schon zu Beginn auf viele Funktionen stösst.

Die Beleuchtung der Insel ist nicht ideal. Einerseits gibt es Bereiche im Nordwesten der Halbinsel, die zu dunkel sind. Andererseits sind manchmal die Formen des Geländes kaum wahrnehmbar, weil alle Hänge gleich beleuchtet werden. Ein Lösungsansatz wäre, eine Beleuchtung einzuführen, die sich mit dem Betrachter bewegt, aber anders als die Stirnlampe von der Seite kommt.

Die einzelnen Kacheln von Methana könnten schon beim Laden gut sichtbar sein. Weil der Farbaustausch für die Geländeoberfläche aber durch Prototypen gelöst ist, werden die Kacheln erst vernünftig dargestellt, wenn alle Dateien geladen sind. Besser wäre hier eine Lösung mit dem Befehl *createVrmlFromURL*. Damit wird es auch möglich, beliebige andere Farbdateien zu verwenden, ohne dass alle Geländekacheln umgeschrieben werden müssen.

Ein kleiner Bug befindet sich in der Menüsteuerung. Wird zweimal der selbe Menüpunkt ausgewählt, wird der Schriftzug mit einer falschen Farbe angezeigt. Zur Lösung des Problems muss für den Austausch der Rastergrafiken eine andere Steuerung geschrieben werden.

11.1.3 Softwareprobleme

Dieser Teil der Schwierigkeiten lässt sich kaum lösen, weil seine Ursachen in der verwendeten Software liegen, also in Netscape, Cortona oder CosmoPlayer.

Hier wäre zu nennen, dass die Viewpoints, die mit den Auswahllisten aufgerufen werden, nicht immer richtig orientiert sind, wenn man sich nach dem letzten Aufruf bewegt hat. Das Problem ist, dass immer der selbe Viewpoint verwendet wird, aber jedes mal neu positioniert wird. Lösen können wir das, indem bei jeder Auswahl ein neuer Viewpoint erzeugt wird.

Manche VMRL-Plug-ins laden keine zusätzlichen Objekte, wenn diese über CGI aufgerufen werden. Damit funktioniert die Datenbankbindung nicht mehr richtig. Die schnellste Lösung war hier, die

Objekte aus Dateien zu laden. Solange sich die Datenbank nicht ändert, hat das keine Auswirkungen.

Weil die Steuerung des EAI mit Javascript geregelt ist, ist Methana 3D nur mit dem Netscape Navigator möglich. Eine Steuerung mit Java könnte das Problem lösen.

Schliesslich hat die Offline-Version die Tendenz, Netscape mit einer Fehlermeldung abzubrechen, wenn man das Hauptfenster von Methana 3D schliesst.

11.1.4 Geschwindigkeitsprobleme

Der letzte Bereich behandelt alle Probleme, die mit einer grösseren Geschwindigkeit gelöst wären.

Um eine Textur über das Gelände zu legen, benötigt der CosmoPlayer ca. zwei Minuten (Cortona schafft das in 20 Sekunden).

Vor allem bei Windows 98 wird das rote Dreieck, das die Ausrichtung des Betrachters zeigt, nur aktualisiert, wenn er in der VRML-Welt die Maus loslässt.

Wird der Viewpoint über die Auswahllisten geändert, flackert das Bild oft störend. Ausserdem wäre es besser, wenn wir sofort sehen könnten, wenn wir uns mit der Übersichtskarte neu positionieren. Ohne Verzögerung wäre die Steuerung mittels zweier Mausklicks leichter zu bemerken.

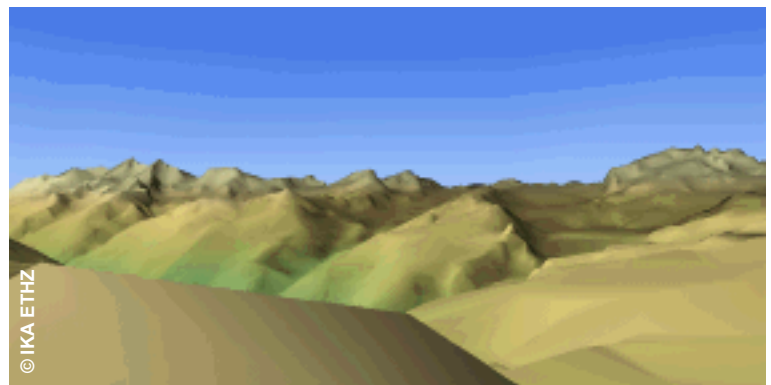


Abbildung 33: Beispiel, das mit LT2VRML erzeugt wurde.

11.2 Erweiterungsmöglichkeiten

Methana 3D ist nur als Prototyp entwickelt. Natürlich sind viele Möglichkeiten zur Erweiterung gegeben.

So sollten alle Hotels und auch noch weitere interessante Gebäude integriert werden. Ob alle Gebäude aufgenommen werden können, ist sicher ein Speicher- bzw. Geschwindigkeitsproblem. Weitere Themen können ebenfalls gefunden werden bzw. vorhandene vervollständigt werden. Dabei kann man sich vorstellen, dass Einzelobjekte wie historische Bauten entweder in die Welt eingebaut werden oder sich in einem neuen Fenster öffnen. Über das Gelände projizierte Luft- oder Satellitenbilder verstärken den realistischen Eindruck deutlich. Mit einem Panoramabild als Hintergrund kann dem Horizont ein natürliches Erscheinungsbild gegeben werden.

Weiter kann Methana 3D in weiteren Sprachen erscheinen, z. B. Deutsch und Griechisch. Teilweise müssten nur die Datenbank ausgebaut und die Abfragen geringfügig erweitert werden.

Weil viele unerfahrene Besucher Schwierigkeiten haben, sich in der virtuellen Welt zu bewegen, können vordefinierte Rundflüge angeboten werden. Interessant wären hier sicher Flüge entlang von Strassen und Wanderwegen. Eine elegante Lösung ist, wenn an jeder Strassenkreuzung ein Wegweiser aufgestellt wird. Wählt man ein Ziel aus, wird man automatisch über die Strasse gefahren, bis man den nächsten Wegweiser erreicht. Dabei können dann noch Informationsseiten angezeigt werden, wenn man an besonderen Orten vorbeikommt. Wenn die Strassenlinien in der Datenbank gespeichert werden, können wir Flüge mit Routingfunktionen verbinden.

Ein andere Ansatz, um Flüge festzulegen ist, die

Möglichkeit, das Digitalisieren von Flugrouten in der Übersichtskarte zuzulassen.

Für ein Informationssystem ist eine Volltextsuche wichtig, mit der wir uns alle Informationen zu einem Begriff auflisten lassen können. Vorteilhaft ist es, wenn wir autorisierten Benutzern Eingabeformulare anbieten, mit denen sie selbst die Datenbank anpassen können und z. B. neue Hotels einfügen können. Eine weitere Diplomarbeit wäre wohl, die Methana 3D in eine plug-in-lose Anwendung zu übertragen. Dazu könnte Java oder Java3D zum Einsatz kommen.

Schliesslich können wir einfache GIS-Funktionen einbauen. Entfernungsmessung und Messungen von Höhenunterschieden verbessern das Verständnis der Halbinsel.

11.3 Übertragung

Um zu zeigen, dass die gefundenen Arbeitsabläufe auf weitere Projekte übertragbar sind, ist der Konverter LT2VRML geschrieben worden. Diese Sammlung von AML- und Perl-Skripten ermöglicht ohne Eingriff ein Kartenblatt aus dem Höhenmodell (DHM 25) des Schweizerischen Bundesamtes für Landestopographie (L+T) in VRML-Kacheln zu zerlegen (Abb. 33). Die komplette Dokumentation findet sich im Anhang.

11.4 Zusammenfassung

In dieser Arbeit wurde aufgezeigt, wie eine touristische 3D-Plattform im Internet aufgebaut werden kann. Es ist gelungen, dies in ansprechender Form

umzusetzen. Sowohl topografische als auch touristische Objekte sind integriert. Die Darstellung des Geländes mit den zusätzlichen 3D Informationen läuft mit einer ausreichenden Geschwindigkeit. Mit Methana 3D können wir uns ein vollständigeres Bild von der Insel machen, als das mit einer zweidimensionalen Darstellung möglich ist. Informationen können auf verschiedene Arten abgerufen werden, und Suchfunktionen sind vorhanden. Durch die Anbindung der Datenbank ist ein schneller Aufbau garantiert und sind einfache Möglichkeiten zur Erweiterung gegeben. Nicht zu vergessen ist der Spass, den man bei den Besuchern beobachten kann. Ihr Interesse wird geweckt und die Zeit erhöht, während der sie sich auf der Webseite aufhalten. Durch weitgehende Automatisierung ist auch die Übertragung auf andere Projekte möglich.

Abbildungsverzeichnis

- Abbildung 1 : Kapstadt mit Kap der Guten Hoffnung, False Bay und Stellenbosch. Gemalt von H. C. Berann. Aus: H. C. BERANN 1986, S. 63, Abb. 6.
- Abbildung 2: Werkzeuge zur 3D Objektbetrachtung im Internet (v. l.): Cult 3D, MetaStream, QuickTime. Aus: <http://www.cult3d.com/gallery/e-commerce/breitling.html> Sep. 1999, <http://www.metastream.com/cgi-bin/gallery.cgi?res=full&model=pot2> Sep. 1999, <http://www.southshore.com/bullshoalsmovies/tower.mov> Sep. 1999.
- Abbildung 3: Screenshot von 3DML. Aus: <http://www.flatland.com/directory/abbott/abbott.3dml> Sep. 1999.
- Abbildung 4: Screenshot von Shout3D. Aus: <http://www.excitextreme.com/fashion/> Nov. 1999.
- Abbildung 5: Screenshot von Java3D. Aus: <http://java.sun.com/products/java-media/3D/index.html> Sep. 1999.
- Abbildung 6: Screenshot von Active Worlds. Aus: <http://www.activeworlds.com/> Nov. 1999.
- Abbildung 7: Screenshot von Entrance. Aus: <http://cybercore.online.de/> Nov. 1999.
- Abbildung 8: Grüner Kegel in VRML.
- Abbildung 9: Das LOD Prinzip (Nach: M. SUTER 1997, S. 60).
- Abbildung 10: Informationsfluss über das EAI.
- Abbildung 11: Road Map von Web3D. Aus: X3D 1999.
- Abbildung 12: Screenshot von «Glasgow Directory ». Aus: <http://iris.abacus.strath.ac.uk/glasgow/> Nov. 1999.
- Abbildung 13: Platzierung von lokalen VRML-Welten auf dem Globus mit einem geoLocation-Knoten. Aus: <http://www.ai.sri.com/VRMLSets/Composite/earth/trees/2/p0p0.wrl> Nov. 1999.
- Abbildung 14: Gelände, das mit einer topografischen Karte als Textur belegt ist. Aus: <http://206.168.216.116/jym/VailTracks/NewVail.wrl> Nov. 1999.
- Abbildung 15: Visualisierung von Geodaten mit VRML (USGS). Aus: <http://virtual.er.usgs.gov/Pinatubo/Pinatubo8.wrl> Nov. 1999.
- Abbildung 16: Screenshot vom traVelleR. Aus: <http://www.geog.le.ac.uk/mek/software/traVelleRTester/traVelleRWebFrames.html> Nov 1999.
- Abbildung 17: Screenshot von der VRML Prism Map. Aus: J. SWANSON 1999.
- Abbildung 18: Methana von Ägina gesehen. Von: Tobias Schorr.
- Abbildung 19: Antike Reste bei der Kapelle Agios Panteleimonas. Von: Tobias Schorr.
- Abbildung 20: Im Innern des Vulkans von Kameni Chora . Von: Tobias Schorr.
- Abbildung 21: Das Dorf Megalochori auf Methana. Von: Tobias Schorr.
- Abbildung 22: Verschiedene Objekte auf Methana 3D.
- Abbildung 23: TIN (Triangulated Irregular Network).
- Abbildung 24: Textur wird auf die maximale Ausdehnung angepasst. Nur die Bereiche mit Polygonflächen werden dargestellt. Wird Pkt 3 mit Pkt 1 gleich gesetzt, ist die Fläche unsichtbar. Die Textur ist nur auf der Kachel sichtbar.

Abbildung 25: In einem VRML-Extrusion-Node sind die Querschnitte nicht immer gleich ausgerichtet.

Abbildung 26: Aufbau der Strassen.

Abbildung 27: Berechnung der Position und Ausrichtung einer Kirche.

Abbildung 29: Infotafeln, die aus der Datenbank erzeugt werden.

Abbildung 28: Aus der Datenbank erzeugte HTML Seite.

Abbildung 30: Hauptfenster von Methana 3D mit Hauptmenü, VRML Welt, Übersichtskarte, Auswahlliste, Inforahmen und Legende (von links nach rechts, von oben nach unten). Aus: <http://www.karto.ethz.ch/~hm/methana/> Nov. 1999.

Abbildung 31: Control-Panel von blaxxun Contact.

Abbildung 32: Vereinfachter Ablauf der Objektanfrage aus der Datenbank.

Abbildung 33: Beispiel, das mit LT2VRML erzeugt wurde. Von: IKA, ETH Zürich 1999.

Literaturverzeichnis

- ARNBERGER, Erik 1993: Thematische Kartographie, mit einer Kurzeinführung über EDV-unterstützte Kartographie und Quellen der Fernerkundung. Braunschweig: Westermann. (= Das Geographische Seminar).
- BERTELSMANN UNIVERSALLEXIKON 1994: Bertelsmann Universallexikon. Das Wissen unserer Zeit von A bis Z. Gütersloh, München. CD-ROM.
- BERANN, Heinrich C. 1986: Darstellende Kunst im Panorama. In: Kartenverwandte Darstellungen Werkstattberichte. Karlsruhe. (= Karlsruher Geowissenschaftliche Schriften. Reihe A: Kartographie und Geographie. Bd. 4, S. 57 - 68).
- BOLLMANN, Friedrich 1986: Entstehung von Bildstadtplänen. In: Kartenverwandte Darstellungen Werkstattberichte. Karlsruhe. (= Karlsruher Geowissenschaftliche Schriften. Reihe A: Kartographie und Geographie. Bd. 4, S. 93 - 98).
- BROCKHAUS 1992: Brockhaus Enzyklopädie, in 24 Bänden. 19. Aufl. Mannheim: Brockhaus.
- BÜRGI, Andreas 1998: Der Blick auf die Alpen: Franz Ludwig Pfiffers Relief der Urschweiz (1762 bis 1786). In: Cartographica Helvetica, H. 18, 1998, S. 3 - 9.
- BUZIEK, Gerd 1997: Das Potential moderner Informations- und Kommunikationstechnologien aus Sicht der Kartographie. In: GIS und Kartographie im multimedialen Umfeld. Grundlagen, Anwendungen und Entwicklungstrends. Bonn: Kirschbaum. (= Kartographische Schriften. Bd. 2). S. 17 - 25.
- CÂMARA, António S. & Jonathan RAPER (Ed.) 1999: Spatial Multimedia and Virtual Reality. London: Taylor & Francis.
- ICA VISCOMM 1999: Commission on Visualization. International Cartographic Association (ICA). <http://www.geog.psu.edu/ica/> . 23.11.1999.
- DUDEN Bd. 8 1986: Duden „Sinn- und sachverwandte Wörter“: Wörterbuch d. treffenden Ausdrücke. 2. Aufl. Mannheim u. a.: Bibliographisches Institut. (= Der Duden: in 10 Bd.: d. Standardwerk zur dt. Sprache. Bd. 8.).
- DYKES, Jason A., Kate MOORE & Dave FAIRBAIRN 1999: From Chernoff to Imhof and Beyond: VRML & Cartography. VRML '99. Paper Submission.
- EAI 1999: External Authoring Working Group, Web3D Consortium. Internet: <http://www.web3d.org/WorkingGroups/vrml-eai/> . 27.9.1999.
- ENNIS, Gareth, Malcolm LINDSAY & Mike GRANT 1999: VRML Possibilities: The Evolution of the Glasgow Model. Paper presented at the Virtual Systems and MultiMedia Conference (VSMM) Dundee, Scotland, September 1999. Internet: <http://iris.abacus.strath.ac.uk/gary/papers/glasgow-model.htm> . 24.11.1999.
- FLANAGAN, David 1998: JavaScript. The Definitive Guide. 3. Aufl. Cambridge u. a.: O'Reilly.
- FLUCKIGER, François 1996: Multimedia im Netz. München u. a.: Prentice Hall Verlag.
- FORBES, Hamish & Christopher MEE (Ed.) 1997: A Rough and Rocky Place: The landscape and settlement history of the Methana peninsula, Greece. Liverpool: Liverpool University Press.
- GEOVRML 1999: GeoVRML Working Group, Web3D Consortium. Internet: <http://www.ai.sri.com/geovrml/> . 22.11.1999
- GEOVRML 1.0 DRAFT 1999: Recommended Practice von GEOVRML 1999. Internet: <http://www.ai.sri.com/geovrml/1.0/> . 20.11.1999
- GIBSON, William 1987: Neuromancer. 8. Aufl. München: Wilhelm Heyne Verlag.

- HAKE, Günter & Dietmar GRÜNREICH 1994: Kartographie. 7., völlig neu bearbeitete und erweiterte Auflage. Berlin, New York: de Gruyter. (= De-Gruyter-Lehrbuch).
- HENNIG, Alexander 1997: Die andere Wirklichkeit. Virtual Reality – Konzepte, Standards, Lösungen. Bonn u. a.: Addison-Wesley-Longman.
- HURNI, Lorenz 1995a: Methana 1:25 000. "Stiftung Vulkaninstitut Immanuel Friedländer" and Institute of Cartography, both at the Swiss Federal Institute of Technology (ETH). Zürich.
- HURNI, Lorenz 1995b: Methana Geological Map 1:25 000. "Stiftung Vulkaninstitut Immanuel Friedländer" and Institute of Cartography, both at the Swiss Federal Institute of Technology (ETH). Zürich.
- HURNI, Lorenz 1995c: Modellhafte Arbeitsabläufe zur digitalen Erstellung von topographischen und geologischen Karten und dreidimensionalen Visualisierungen. DISS. ETH Nr. 11066. Zürich.
- KAGIA, Betty 1995: Die Inseln des Saronischen Golfes. Athen: Grecocard-Verlag.
- METHANA o. J.: Information Guide. Griechisch, Englisch. CD-ROM. Piräus.
- MOORE, Kate 1999: VRML and Java for Interactive 3D Cartography. In: Cartwright, William et al. (Ed.): Multimedia Cartography. Berlin u. a.: Springer, 1999, S. 205 – 216.
- NADEAU, David R. 1999: Introduction to Programming with Java 3D. In: SIGGRAPH 99. 08–13 August 1999. Los Angeles, California. Course Notes. Courses Committee. CD-ROM.
- REDDY, Martin & Lee IVERSON 1999: Enabling Geographic Support in Virtual Reality Modeling with GeoVRML. Note 13. In: Cartography and Geographic Information Science. Vol. 26 (1999), No. 3, S. 180.
- REDDY, Martin, Yvan LECLERC et al. 1999: TerraVision II: Visualizing Massive Terrain Databases in VRML. In: IEEE Computer Graphics & Applications, Vol. 19, No. 2, March/April 1999, S. 30 - 38. (= Internet: <http://www.ai.sri.com/~reddy/pubs/CGAVRML/> . 23.11.1999).
- RHYNE, Theresa-Marie 1999: Summary of August 11, 1999 - Carto BOF Meeting at SIGGRAPH 99. Internet: <http://www.siggraph.org/~rhyne/carto/carto99sum.html> . 23.11.1999.
- SCHLÜTER, Oliver 1998: VRML. Sprachmerkmale, Anwendungen, Perspektiven. Köln: O'Reilly.
- SCHORR, Tobias 1996: Methana. Die Vulkanhalbinsel Methana. Der Reiseführer über die Geologie, Natur und Geschichte. Unveröffentlicht. 26. 7. 1996.
- SUTER, Martin 1997: Aspekte der interaktiven real-time 3D-Landschaftsvisualisierung. (= Remote Sensing Series. Vol. 29). Zürich.
- SWANSON, James 1999: The Cartographic Possibilities of VRML. In: Cartwright, William u. a. (Ed.): Multimedia Cartography. Berlin u. a.: Springer, 1999, S. 181 – 194.
- X3D 1999: Extensible 3D Task Group, Web3D Consortium. Internet: http://www.web3d.org/fs_x3d.htm . 27.9.1999.
- ZANINI, Marc 1998: Dreidimensionale synthetische Landschaften. Wissensbasierte dreidimensionale Rekonstruktion und Visualisierung raumbezogener Informationen. Zürich: ETH Zürich. (= Institut für Geodäsie und Photogrammetrie an der ETH Zürich. Mitteilungen Nr. 66).
- ZEDI, Roger 1998: Dynamische und interaktive Darstellung geografischer Objekte. Internet: <http://www.geo.unizh.ch/~zedi/diplom/> . 5.10.1999.
- ZEHNDER, Carl August 1987: Informationssysteme und Datenbanken. 4. Aufl. Zürich: Verlag der Fachvereine an den Schweizerischen Hochschulen und Techniken.