3D-Online-Avatar

Technische Konzeption und Realisierung eines Avatar-Systems zum Einsatz im Internet

Diplomarbeit an der
Fachhochschule Stuttgart – Hochschule der Medien
Fachbereich Electronic Media
Studiengang Audiovisuelle Medien

FACHHOCHSCHULE STUTTGART
HOCHSCHULE DER MEDIEN

vorgelegt von

Tobias Scheck
Matrikelnummer 10895
tobias.scheck@topazmedia.de

im Oktober 2002

1. Prüfer: Prof. Dr. Johannes Schaugg
2. Prüfer: Prof. Uwe Schulz

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Diplomarbeit mit dem Titel

3D-Online-Avatar

Technische Konzeption und Realisierung eines Avatar-Systems zum Einsatz im Internet

selbständig und ohne fremde Hilfe verfasst habe. Ich habe dazu keine weiteren als die angeführten Hilfsmittel benutzt und die aus anderen Quellen entnommenen Stellen als solche gekennzeichnet.

Stuttgart, den 1. Oktober 2002

Tobias Scheck

Inhaltsverzeichnis

Abbildungsverzeichnis

Abkürzungsverzeichnis

API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
ASF	Advanced Streaming Format
AVI	Audio Video Interleave
CGI	Common Gateway Interface
DTD	Document Type Definition
GUI	Graphical User Interface
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer/Transmission Protocol
ISDN	Integrated Services Digital Network
MP3	MPEG-1 Layer 3
MPEG	Moving Picture Experts Group
ODBC	Open Database Connectivity
RCSL	RealNetworks Community Source License
RPSL	RealNetworks Public Source License
RTP	Real Time Protocol
SAPI	Speech Application Programming Interface
SDK	Software Developer's Kit
TTS	Text-To-Speech
VET	Viewpoint Experience Technology
VMP	Viewpoint Media Player
VRML	Virtual Reality Modeling Language
WAV	Wave Length Encoding
X3D	Extensible 3D
XML	Extensible Markup Language

1. Einleitung

Avatare – der Begriff stammt aus dem Hinduismus und bezeichnet wiedergeborene Wesen, die auf die Erde herabsteigen, um die bedrohte Weltordnung zu schützen. Der Begriff wurde Anfang 1980 in den Computerbereich übertragen, als Programmierer des US-Militärs nach einem Begriff für die menschlichen Artefakte in ihren Simulationsprogrammen suchten. Die heute wohl berühmtesten virtuellen Identitäten sind Lara Croft, Tomb Raider und Kyoko Date, die aus dem 3D- bzw. Computerspiel-Bereich stammen.

Der Avatar-Boom, speziell im Internet, befindet sich erst ganz am Anfang. Dennoch erfreuen sich künstliche Figuren und virtuelle Berater gerade in diesem Medium zunehmender Beliebtheit. *„Rund 40 Prozent der deutschen Internetnutzer wünschen sich Avatare“* [1] - zu dieser Einschätzung kommt die Unternehmensberatung Mummert + Partner. *„Was die Multimedia- und Internetwelt heute mit Avataren auf die Beine stellt, ist erst der Anfang einer Entwicklung, die in den kommenden Jahren weiter an Dynamik gewinnen wird“* [2] prognostiziert Uta Springer von der comvos online medien GmbH. Offensichtlich ist, dass der Service am Kunden im Bereich des E-Commerce noch stark verbesserungsfähig ist, und dass die Entwicklung von Avataren ein Schritt in diese Richtung ist. Viele Online-Shops gleichen heute den Geisterstädten aus alten Wild-West-Filmen. Angebote in Hülle und Fülle – wird jedoch zeitnaher Rat gebraucht, stehen Benutzer auf verlorenem Posten.[3]

E-Commerce ist also ein Bereich in dem Avatare als virtuelle Kundenberater einen guten Service leisten könnten. Es gibt allerdings weitere Anwendungsfelder, wie zum Beispiel E-Learning, Content-Providing (News) oder Entertainment, in denen der Einsatz von Avataren großen Nutzen bringen kann. Die aktuelle Marktsituation sieht hingegen so aus, dass Avatare im Internet nur sehr vereinzelt anzutreffen sind. Das liegt zum einen an zögerlichen Betreibern von Webangeboten, aber auch an der noch unausgereiften Technologie in diesem Bereich.

Die Entwicklung von Avatar-Systemen im Internet steckt noch in den Kinderschuhen. Gerade deshalb ist die Diplomarbeit in diesem Bereich angesiedelt, der noch viel Raum für neue Ideen und Technologien lässt.
Die Arbeit besteht aus zwei Teilen, einem theoretischen Teil und einem praktischen. Im ersten Teil soll ein Überblick über Anforderungen an ein Avatar-System, in Frage kommende Systemkomponenten, sowie denkbare Systemarchitekturen gegeben werden. Im zweiten, dem praktischen Teil, wird die Implementierung zweier unterschiedlicher Avatar-Systeme beschrieben, die im Rahmen dieser Diplomarbeit implementiert wurden.

[1] Mummert + Partner Unternehmensberatung Aktiengesellschaft: Virtuelle Assistenten: Geheimwaffe gegen Milliardenverluste im Internet.
Online in Internet: „URL: http://www.mummert.de/deutsch/press/a_press_info/01030a.html [Stand: 24.07.2002]“.
[2] Springer, Uta: Avatar, vertritt mich!. Newsletter 12/2002 der comvos online medien GmbH.
Online in Internet: „URL: http://www.comvos.de/media/pdf/newsletter1200.pdf [Stand: 31.07.2002]“.
[3] Vgl. Robben, Matthias: Service Sells – Kundenberatung online.
Online in Internet: „URL: http://www.ecin.de/strategie/kundenberatung/ [Stand: 24.07.2002]“.

2. Grundlagen Avatar-Systeme

„Ein Avatar ist eine fiktive Identität, die im Internet oft die Funktion der Hilfe übernimmt. Als Nachahmung realer Menschen oder anderer Lebewesen führt er den Nutzer durch das Angebot“ [1] – so eine aktuelle Definition für den Begriff Avatar. Die Verwendung dieses Begriffes ist allerdings sehr vielseitig und so wird er für viele verschiedene Dinge verwendet. Es werden Assistenten für Anwendungsprogramme, wie beispielsweise der Microsoft Office-Assistent, Icons in Online-Chats, die Benutzer repräsentieren, autonome Programme, wie Biet-Agenten in Internet-Auktionshäusern[2] oder aber virtuelle Models, die Mode in Online-Shops vorführen, als Avatare bezeichnet.

All diesen Erscheinungen ist eines gemeinsam: sie sollen Benutzern helfen, indem sie aktiv werden und dem Benutzer Arbeit abnehmen. Die Arbeit, die von Avataren erledigt wird, besteht dabei im Auffinden von Information, der Erledigung bestimmter Tätigkeiten aufgrund vorher definierter Umstände oder der Präsentation von Information, Produkten und vielem mehr. Von einem Avatar wird also erwartet, dass er ein gewisses Maß an Eigeninitiative und Aktivität zeigt, ja sogar intelligent ist.

Künstliche Intelligenz – dieser Begriff ist jedoch nur eingeschränkt auf Avatare, wie sie im Internet vorkommen, anzuwenden. Zwar sollen die Avatare fähig zur Kommunikation, also im Stande sein, Fragen zu erkennen und darauf sinnvoll zu antworten, unter künstlicher Intelligenz versteht man jedoch noch mehr. Systeme, die künstliche Intelligenz aufweisen, arbeiten in aller Regel mit einer sogenannten Wissensbasis. Diese Wis-

sensbasis stellt strukturiertes Wissen, oft zu einem speziellen Bereich, dar, welches das System beispielsweise dazu verwendet, Antworten auf gestellte Fragen zu finden. Dieser Anforderung werden auch die hier behandelten Avatare genügen. Künstliche Intelligenz beinhaltet jedoch noch einen weiteren Schritt. Ein „intelligentes“ System soll nicht nur in der Lage sein, eine vorhandene Wissensbasis zur Beantwortung von Fragen zu benützen, es soll diese Wissensbasis durch Kommunikation sukzessive vergrößern können, also neues Wissen akquirieren. Wissensakquisition bedeutet Strategien zu entwerfen, wie Programme zu benutzerunabhängigen Lernprozessen befähigt werden können.[3]

Dieses Feld wird von Avataren im Internet bisher nicht abgedeckt. Man beschränkt sich bei heutigen Avatar-Systemen auf die Verwendung einer statischen Wissensbasis, die nicht selbständig durch den Avatar vergrößert werden kann.

2.1. Was ist ein 3D-Online-Avatar?

Was ist nun aber unter einem 3D-Online-Avatar, dem virtuellen Charakter, wie er in einem Avatar-System im Internet verwendet wird, zu verstehen?

Der Zusatz „3D“ weist schon auf die Form des Charakters hin, er liegt dreidimensional vor. Im Gegensatz zu verbreiteten Formaten, wie beispielsweise Macromedia Flash, die zweidimensional arbeiten, liegt hier also eine wirkliche 3D-Figur vor, die im Browser des Benutzers gerendert wird. Es ist somit möglich, den Charakter von allen Seiten zu betrachten und ihn im virtuellen Raum zu animieren. Die Erscheinung der Figur ist wesentlich plastischer als 2D-Formate das leisten können, das haptische Erlebnis[4] des Benutzers wird gesteigert. Die weitere Bezeichnung „Online“ deutet auf die Verwendung des Avatar-Systems im Internet hin. Es handelt sich also um einen netzwerkfähigen Avatar, der über ein Computernetzwerk, wie das Internet oder ein Intranet, mit dem Benutzer kommunizieren kann. Der Begriff „Avatar“ wurde schon kurz eingeführt. Man versteht eine

fiktive Figur darunter, die dem Benutzer helfen, ihm verschiedene Arbeiten bzw. Tätigkeiten abnehmen soll.

Der 3D-Online-Avatar ist also eine virtuelle Figur, als 3D-Modell erstellt, die im Internet agiert und fähig ist, mit dem Benutzer zu kommunizieren, ihm auf Fragen, vielleicht zu einem speziellen Gebiet, passende Antworten zu geben.

2.2. Aktueller Stand der Technik

3D-Online-Avatare sind zum aktuellen Zeitpunkt noch eine recht exotische Erscheinung im Internet, deren Verbreitung als eher gering einzustufen ist. Die Gründe hierfür sind vielseitig, hauptsächlich liegt es jedoch an der technischen Unreife der Systeme und dem damit einhergehenden Mangel an Flexibilität.

Viele 3D-Systeme, wie beispielsweise auch ThreeDee[A] oder Victor[B] der Firma Egisys, arbeiten nach einem sehr statischen Konzept. Alle Animationen des Avatars sind hier fest mit bestimmten Antworten verknüpft, die Sprachausgabe wird über das Abspielen von mp3-Dateien realisiert. Diese mp3-Dateien müssen zuvor von Sprechern im Tonstudio aufgenommen werden, der Aufwand ist dementsprechend hoch. Problematisch ist, dass eine Erweiterung der Wissensbasis, also das Hinzufügen neuer Antworten, immer die Neuproduktion von Audiomaterial mit sich bringt. Aufwand und Kosten steigen so sehr schnell in eine Höhe, die den Avatar unrentabel macht. Die Aktualität der Wissensbasis ist jedoch ein essentieller Bestandteil, der für den Erfolg eines Avatars unverzichtbar ist. *„Außerdem muss ein System ständig aktualisiert werden, damit keine neuen Fragen dazukommen, auf die das System noch keine Antwort hat“* [5] – das weiß auch Ralf Raule, freischaffender Entwickler für die KiwiLogic-Systeme. Ki-

[A] ThreeDee ist online im Internet zu sehen: „URL: http://www.egisys.de/demo/Avatare/ThreeDee/ [Stand: 31.07.2002]“.
[B] Victor ist online im Internet zu sehen: „URL: http://www.egisys.de/demo/Avatare/Victor/ [Stand: 31.07.2002]“.

wilogic ist ein Hamburger Unternehmen, welches das LinguBot-System entwickelt hat. Dieses System ist nach Angaben des Unternehmens das meistverkaufte natürlichsprachliche Dialogsystem der Welt.[6]

Die meisten Avatar-Systeme im Internet verzichten daher bislang vollständig auf 3D-Visualisierung und Audio. Die Antworten werden in einer HTML-Seite als Text ausgegeben, der Avatar wird durch eine Grafik dargestellt, die entweder das Foto eines realen Menschen oder aber die Abbildung einer fiktiven Figur zeigt.

2.3 Einsatzgebiete

Die Einsatzgebiete für Avatar-Systeme sind vielseitig. Bereits angesprochen wurde der Einsatz im E-Commerce-Bereich. Hier sollen Avatare als virtuelle Verkäufer auftreten und die Umsatzzahlen der Online-Shops steigern. So will beispielsweise auch Bertelsmann Online in seinem Web-Buchgeschäft Avatare als Verkäufer einsetzen.[7]

„Die Finanzwirtschaft setzt auf Avatare, vor allem um die Hemmschwelle des Normalbürgers vor der Online-Filiale zu senken“ [8] – auch Banken und Versicherungen sehen also ein großes Potential in dieser neuen Technologie.

Ein weiteres Geschäftsfeld für Avatare begründet die britische Nachrichtenagentur Press Association mit ihrer virtuellen Nachrichtensprecherin „Ananova“ und auch das ZDF hat schon mit seinem Webface „Cornelia“ erste Gehversuche im Internet gemacht.[9]

Ein beeindruckendes, wenn auch für europäische Verhältnisse etwas grotesk wirkendes Beispiel ist der große Triumph von Kyoko Date, die 1996 die virtuelle Bühne betrat, um schon kurze Zeit später ein umjubeltes Popidol in Japan zu sein.

Showbusiness und Nachrichtenbereich sind also ebenso potentielle Einsatzgebiet wie Entertainment, Dienstleistung oder Verkauf.

[1] Koordinierungs- und Beratungsstelle der Bundesregierung für Informationstechnik in der Bundesverwaltung im Bundesministerium des Innern: Glossar.
Online in Internet: „URL: http://www.bund.de/BundOnline-2005/SAGA/Glossar-.6343.htm [Stand: 29.07.2002]“.
[2] Vgl. Krempl, Stefan: Zeigt her eure Pixel.
Online in Internet: „URL: http://www.heise.de/tp/deutsch/inhalt/te/7697/1.html [Stand:29.07.2002]“.
[3] Vgl. Becker, Barbara: Künstliche Intelligenz: Konzepte, Systeme, Verheißungen.
Frankfurt/Main, New York 1992, S. 21.
[4] Koglin, Ilona: Visuell greifbar.
In: Page, 03/2002, S. 72.
[5] Raule, Ralf, zitiert nach Puscher, Frank: Bau dir deinen Avatar.
In: Internet World, Februar 2002, S. 86.
[6] Vgl. KiwiLogic: Kiwilogic Lingubot: Das meistverkaufte natürlichsprachliche Dialogsystem der Welt.
Online in Internet: „URL: http://www.kiwilogic.de/kiwilogic/site/_xml/cont_index.php?menue_id=10055&submenue_id&id [Stand: 15.08.2002]“.
[7] Vgl. Springer, Uta: Avatar, vertritt mich!. Newsletter 12/2002 der comvos online medien GmbH.
Online in Internet: „URL: http://www.comvos.de/media/pdf/newsletter1200.pdf [Stand: 31.07.2002]“.
[8] Ebenda.
[9] Vgl. ebenda.

3. Systemanforderungen

Vor der Planung eines Avatar-Systems gilt es die Anforderungen an ein solches genau zu definieren. Die Frage ist: Was muss das System können?

Da das System im Internet eingesetzt werden soll, müssen auf jeden Fall einige Bedingungen, die speziell dieses Medium mit sich bringt, beachtet werden. Im Internet bewegen sich unzählige Benutzer mit zahlreichen unterschiedlichen Computersystemen. Es gibt verschiedene Betriebssysteme, wie Microsoft Windows, Apple Macintosh oder Linux. Außerdem kommen unterschiedliche Browser, wie der Internet Explorer, Netscape oder Mozilla zum Einsatz, wobei Browser Plug-ins meist nur zu bestimmten dieser Browser kompatibel sind. Einige der Plug-ins sind auf vielen Systemen bereits standardmäßig installiert, andere muss der Benutzer erst herunterladen und dann installieren, was oft ein Hindernis darstellt. Es ist also anzustreben, einen möglichst hohen Anteil der Benutzer zu erreichen, was heißt eine möglichst große Zahl unterschiedlicher Systemkonfigurationen zu unterstützen.

Ein wichtiger Aspekt ist die 3D-Darstellung des Avatars. Um einen plastischen Charakter zu erzielen, muss der Charakter auf jeden Fall als dreidimensionales Model im Browser des Benutzers gerendert werden. Kunden werden durch die dreidimensionalen Wesen in besonderer Weise angesprochen[1], das meint Kai Bühler, Vorstandsvorsitzender der Kölner Fir-

ma plan_b media[A]. Da die Darstellung von 3D-Objekten nicht standardmäßig durch Internetbrowser möglich ist, muss hier auf zusätzliche Plug-ins zurückgegriffen werden. Kriterien, wonach die Plug-ins untersucht werden müssen, sind deren Verwendbarkeit für das System, also ob die technischen Vorraussetzungen überhaupt erfüllt sind, sowie Qualität und Verbreitung.

Ein essentieller Aspekt für die Funktionsfähigkeit eines Avatar-Systems ist dessen Fähigkeit zur Kommunikation. Es muss ein Dialogsystem integriert werden, das in der Lage ist, gestellte Fragen zu erkennen und aufgrund einer vorhandenen Wissensbasis passende Antworten zu geben. Der Wortschatz und die Wissensbasis sollen möglichst einfach und kostengünstig verwaltet werden können. Neu zu erkennende Fragen und damit verknüpfte Antworten müssen jederzeit ins System eingespeist werden können. Eine Art Content Management System wird somit für das Wissen des Avatars benötigt. Die Pflege des Wissens erfolgt auf Textbasis, da die Erstellung und Pflege von Audiodateien zu kostenintensiv und aufwändig ist.

Die Antworten des Avatars sollen in jedem Fall als gesprochene Sprache, also Audio ausgegeben werden. Da die Wissensbasis auf Text basiert und die Antworten somit ebenfalls in Textform vorliegen, kommt die Ausgabe durch im Vorfeld produzierte Audiodateien nicht in Frage. Ein Text-To-Speech-System (TTS) wird benötigt. Dieses System ist in der Lage, Text in gesprochene Sprache umzusetzen, Audio kann somit dynamisch aufgrund der aktuellen Wissensbasis erzeugt werden. Wird ein TTS-System auf Serverseite eingesetzt, so bedarf es zur Ausgabe des Tons im Browser des Benutzers allerdings eines geeigneten Plug-ins. Alternativ gibt es auch die Möglichkeit, ein TTS-System auf dem Computer des Benutzers zu verwenden, doch auch dann muss ein Plug-in vorliegen, welches dieses System ansteuern kann.

[A] plan_b media ist ein Kölner Medienunternehmen, spezialisiert auf Entwicklung, Gestaltung und Vermarktung interaktiver 3D-Charaktere. plan_b hat den populären Charakter Baby Fred entwickelt.

Zusammenfassend lassen sich folgende Punkte feststellen, die das zu entwickelnde Avatar-System abdecken sollte:

- Hohe Browser- und Systemkompatibilität
- 3D-Darstellung im Browser
- Qualitativ hochwertiges bzw. verbreitetes 3D-Plug-in
- Content Management System zur Pflege der Wissensbasis
- Dialogsystem für Kommunikationsfähigkeit
- Natürliche Sprachausgabe über TTS
- TTS client- oder serverseitig
- Audioausgabe über Plug-in bei serverseitigem TTS

Diese Punkte müssen bei der Auswahl der Systemkomponenten und der Konzeption der Systemarchitektur berücksichtigt werden. Nur so ist gewährleistet, ein System zu schaffen, das möglichst vielen Ansprüchen gerecht wird und auch marktfähig ist.

[1] Vgl. Bühler, Kai, zitiert nachTang, Miriam: Virtuelle Figuren sollen das Web “persönlicher” machen.
Online in Internet: „URL: http://www.heise.de/newsticker/result.xhtml?url=/newsticker/data/cp-01.07.01-001/default.shtml&words=Avatar [Stand: 01.08.2002]“.

4. Systemkomponenten

Bei der Auswahl der richtigen Systemkomponenten für ein Avatar-System spielen viele Faktoren eine Rolle. So ist wichtig zu analysieren, welche Technologien in der Lage sind zusammenzuarbeiten. Wo gibt es bereits Schnittstellen, wo kann man eigene Schnittstellen definieren – das sind Fragestellungen, mit denen man sich hier oft konfrontiert sieht. Bei der Auswahl von Plug-ins ist weiter zu untersuchen, wie verbreitet diese sind, welchen Aufwand die Installation bedeutet, wie die Zukunftschancen einer Technologie abzuschätzen sind und wie hoch die Qualität einzustufen ist.

4.1. Kommunikation

Für Kommunikation zwischen User und Avatar ist ein sogenanntes Dialogsystem notwendig. Auf dem Markt gibt es zahlreiche dieser Systeme, wie z.B. Clarity[A] oder auch CreaLog[B], die stärker im Bereich der Telefonie zu finden sind. Diese Systeme arbeiten häufig mit Spracherkennung, was jedoch für das Internet kaum praktikabel ist. Nicht jeder Benutzer hat ein Mikrofon an seinen Computer angeschlossen und auch der Zugriff auf diese Hardware-Ressource ist mit zahlreichen Komplikationen behaftet. Die Erfassung von Fragen durch HTML-Formulare bietet hier mehr Betriebssi-

[A] Clarity ist ein Unternehmen mit Sitz in Bad Homburg. Das Kerngeschäft von Clarity liegt in der Entwicklung und dem Vertrieb von Softwareprodukten sowie der Erbringung von Professional Services im Bereich multimedialer Sprach-Dialogsysteme und entsprechender Plattformen (URL: http://www.clarity-ag.net).

[B] Die Firma CreaLog hat Standorte in München, Frankfurt und Wien. Das Unternehmen stellt u. a. Voice Portale, also die Verbindung zwischen Telefonie und Datenverarbeitung her (URL: http://www.crealog.de).

cherheit und auch die Bedienung stellt sich für den Benutzer leichter dar, da er diese Form der Kommunikation bereits von zahlreichen Seiten gewohnt ist. Das System, welches für den Einsatz im Internet die meisten Vorzüge bietet und so wohl mit Abstand das Verbreitetste in diesem Bereich ist, ist das LinguBot System. Wie bereits erwähnt, ist die Software LinguBot der Firma KiwiLogic das nach eigenen Angaben meistverkaufte natürlichsprachliche Dialogsystem der Welt.

4.1.1 KiwiLogic LinguBot

KiwiLogic's Lingubot besteht aus zwei Komponenten, dem LinguBot Creator und der LinguBot Web Engine.

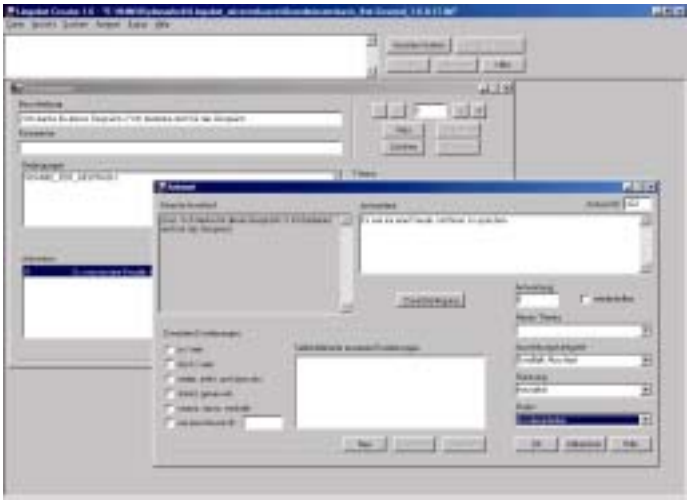
Abbildung 1: KiwiLogic LinguBot Creator

Der LinguBot Creator ist eine Windows-Applikation, die zur einfachen und schnellen Erstellung von Wissensbasen dient. Über eine grafische Oberfläche werden Antworten in die Wissensbasis eingepflegt. Per Definition von Bedingungen kann eine Antwort dann möglichen Fragen zugeordnet werden. Die Bedingung für die Antwort „Hallo, wie geht es Ihnen heute“ könnte zum Beispiel so aussehen: `guten&tag`. Auf eine Anrede, welche die Wörter „guten“ und „tag“ enthält, würde der LinguBot dann eben genau diese Antwort ausgeben. Dieses Beispiel ist nun sehr simpel und würde häufig zu Verwechslungen führen. So würde der Bot hier auch auf die folgende Formulierung obenstehende Antwort ausgeben: „Mir geht es schlecht, ich habe schon seit Jahren keinen guten Tag mehr gehabt.“ Man sieht, es ist nicht einfach, Bedingungen zu definieren, die alle möglichen Formulierungen in

richtiger Art und Weise erfassen. Daher sehen die Bedingungen meist komplizierter aus. Die Bedingung, die beispielsweise als Antwort „Es war mir eine Freude, mit Ihnen zu sprechen“ ausgibt, lautet:

```
%GESPRÄCH&für&(danke/dankeschön/dankesehr/((herzlichen/
schönen/vielen/besten/aufrichtigen/innigsten/wärmsten/
tausend/hab/habt/(haben&sie))&dank)/(man&dankt)/
(ich&(bedanke/bedank)&mich)/schankedön/thanks/thanx/(th
ank+you))&!(kein/null)
```

Da die Übersicht bei solch komplexen Bedingungen schnell verloren ginge, gibt es die Möglichkeit, Bedingungen als Makros abzuspeichern und diese dann mit der Syntax `%MAKRONAME` zu referenzieren. Ist die Wissensbasis fertiggestellt, wird der LinguBot veröffentlicht und auf einem Webserver installiert. Standardmäßig besteht die Oberfläche des LinguBots im Internet aus einer HTML-Seite. Diese Seite enthält ein Formular zum Abschicken der Fragen des Benutzers, einen Textbereich, wo die Antworten ausgegeben werden und ein zur jeweiligen Antwort passendes Bild. Für jede Antwort kann im LinguBot Creator eine Stimmung gewählt werden, wie etwa freundlich, ernst oder fragend. Aufgrund dieser Stimmung wird dann ein passendes Bild auf der Seite angezeigt, der Bot wirkt so, als ob er emotional reagiert.

Die Serverapplikation, die für die Kommunikation zuständig ist, ist die LinguBot Web Engine. Es handelt sich dabei um ein Programm, das für die Serverbetriebssysteme Windows, Linux und Solaris verfügbar ist. Aufgabe dieses Programms ist, die Fragen des Benutzers zu parsen, diese gegen die in der Wissensbasis definierten Bedingungen zu vergleichen und die passenden Antworten auszugeben. Das Programm kommuniziert mit dem Webserver über die CGI-Schnittstelle. CGI steht für Common Gateway Interface, es handelt sich dabei um eine standardisierte Programmierschnittstelle zum Datenaustausch zwischen Browser und Programmen auf dem Webserver. Die LinguBot Web Engine protokolliert sämtliche Gesprächsverläufe in Loggdateien, die später im Creator ausgewertet werden können.[1] Weiter stellt die Web Engine einige Schnittstellen zur Anbindung an andere Systeme zur Verfügung. So lassen sich Shop-Systeme, wie In-

tershop oder OpenShop an den LinguBot anbinden, die Engine kann über die Schnittstellen Open Database Connectivity (ODBC) oder Java Database Connectivity (JDBC) mit Datenbanken kommunizieren oder aber es können weitere Services, wie Aktienkurse oder Wetternachrichten per HTTP-Schnittstelle zugänglich gemacht werden.[2] Ein Vorteil der Web Engine ist, dass ihr Output auf Templates basiert. Die Web Engine befüllt diese Templates nach jeder Anfrage mit den Ausgabe-Daten, wie zum Beispiel dem Ausgabetext. Von Vorteil ist, dass die Templates angepasst werden können, auch eigener JavaScript[C]- Code kann in diese Tempates eingefügt werden. Dies birgt die Möglichkeit, später eine Schnittstelle zu einem TTS-System oder Plug-in in JavaScript zu definieren.

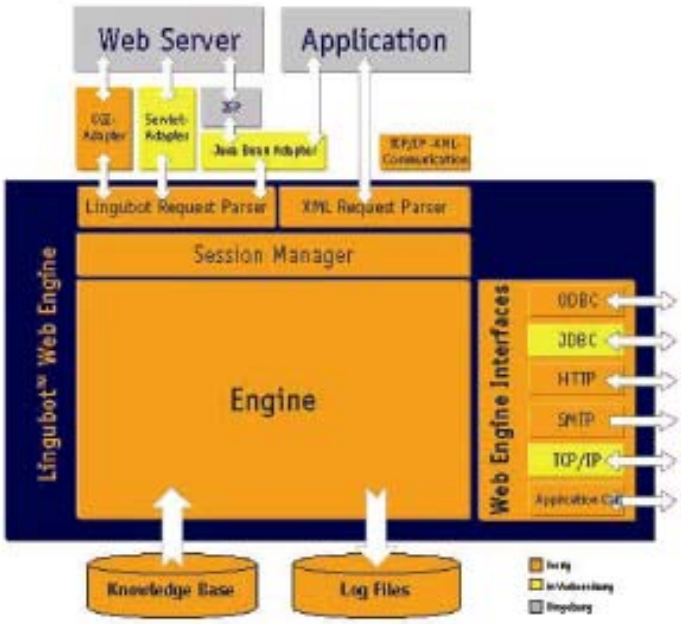

Abbildung 2: KiwiLogic LinguBot Web Engine[3]

Die große Stärke des Kiwilogic LinguBot liegt darin, dass die Software speziell für die Verwendung im Internet auf einem WWW-Server konzipiert ist. Viele benötigte Funktionalitäten, wie z.B. der bereits erwähnte auf Templates basierende Output, stellt das System so von Haus aus zur Verfügung. Zudem lässt sich der Inhalt von Wissensbasen über die grafische Oberfläche des LinguBot Creator verwalten, ein ausgereiftes Werkzeug für Content Management steht damit zur Verfügung. Die Qualität der erzeugten Dialoge, also die der Erkennung von Fragen und Ausgabe von Antworten, wird ausschließlich durch die Komplexität der Wissensbasis bestimmt.

[C] JavaScript ist eine ursprünglich von der Firma Netscape Communications Corporation definierte und am meisten verbreitete Skriptsprache zur Verknüpfung von Programmcode mit statischen HTML-Seiten.

Allgemeine Elemente eines Gesprächs, wie beispielsweise Smalltalk, werden in vorgefertigten Wissensbasen zur Verfügung gestellt. Spezielle Gesprächsthemen können selbständig ins System eingepflegt werden. Durch die Verwendung von Makros wird eine effiziente Erkennung von Fragen erreicht. Um eine qualitativ hochwertige Wissensbasis für ein Spezialgebiet zu erstellen, muss mit einer nicht unerheblichen Entwicklungszeit gerechnet werden. Laut Björn Gülsdorff, Leiter des Autoren-Teams bei KiwiLogic, braucht ein erfahrener Autor etwa drei Monate zur Erstellung einer komplett neuen Wissensbasis.[4] Dafür ist dann aber auch davon auszugehen, dass die Erkennung von Fragen und Ausgabe von Antworten gut funktioniert. Die Verwendung des KiwiLogic LinguBot als Dialogsystem für einen 3D-Online-Avatar erscheint sinnvoll, die notwendigen Vorraussetzungen sind erfüllt .

4.2. Audio-Erzeugung

Ein 3D-Online-Avatar soll sprechen, der Klang der Sprache soll sich so natürlich wie möglich anhören. Am natürlichsten hören sich natürlich wirklich gesprochene Sprecheraufnahmen aus einem Tonstudio an. Da alle Antworten des Avatars aber in Textform vorliegen und diese ständig aktualisierbar sein sollen, muss hier auf ein TTS-System zurückgegriffen werden, ein System, das geschriebenen Text in gesprochene Sprache umsetzt. Man unterscheidet zwischen clientseitigem und serverseitigem TTS, auf die Unterschiede wird später genauer eingegangen. Der Vorgang aus Text Sprache zu erzeugen, ist sehr komplex, was darin begründet ist, dass Sprache schon an sich eine sehr komplexe Sache ist. Um die Funktionsweise eines TTS-Systems zu verstehen, sind daher grundlegende Kenntnisse über Aufbau und Erzeugung von Sprache notwendig. Nur so ist es möglich, eigene Schnittstellen zu einem TTS-System zu definieren.

4.2.1. Grundlagen Sprache und Text-To-Speech

Analysiert man ein Gespräch zwischen zwei Menschen, so lässt sich eine Verkettung physiologischer und kausaler Abläufe feststellen, man spricht von einer lautsprachlichen Kommunikationskette[6]. Die Abläufe lassen sich folgendermaßen gliedern. Zu Beginn entsteht die Sprache im Gehirn des Sprechers. Auf dieser sogenannten linguistischen Ebene wird die Sprache, die an den Kommunikationspartner übermittelt werden soll, im Gehirn des Sprechers als vollständige und zusammenhängende Sätze erzeugt und bereits dort in sogenannte Phoneme, kleinste Sprachelemente, aufgeteilt. Anschließend werden diese Phoneme durch die Sprechorgane in Laute, auch Phone genannt, umgesetzt. Hier spricht man von der physiologischen Ebene. Auf akustischer Ebene wird die Sprache dann in Form von Schallwellen über den Kanal Luft zum Kommunikationspartner übertragen. Bei diesem finden dieselben Vorgänge in umgekehrter Reihenfolge statt. Das Gehör nimmt die Schallwellen auf und wandelt sie in Phone um. Das Gehirn erschließt die Bedeutung und den Sinn der Sprache, indem es die Phone wieder in Phoneme umwandelt und diese schließlich zu den ursprünglichen Sätzen und Satzgefügen zusammensetzt.

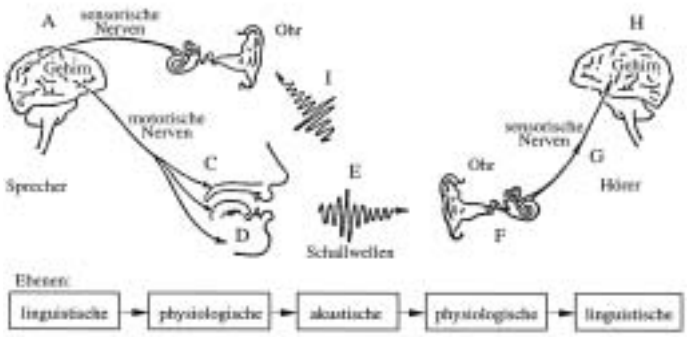

Abbildung 3: Die lautsprachliche Kommunikationskette[5]

Man unterscheidet zwischen der Sprache als System (linguistische Ebene) und der Realisierung der Sprache (physiologische Ebene).[7] Die kleinsten Elemente der Sprache sind dabei auf linguistischer Ebene die Phoneme, auf physiologischer Ebene sind es die Phone. Die Zahl der Phoneme, die in einer Sprache vorkommen, ist verhältnismäßig klein. So existieren im Deutschen etwa 50 Phoneme.[8]

Die Kommunikationskette, wie sie sich in einem Avatar-System darstellt, sieht hingegen etwas anders aus. In diesem System wird je nach Richtung der Kommunikation unterschiedlich kommuniziert. Der Benutzer kommuniziert mit dem Avatar auf textueller Ebene, d.h. er stellt Fragen durch die Eingabe von Text. In Textform gestellte Fragen sind wesentlich einfacher zu verarbeiten, Audioanalyse und Spracherkennung entfallen. Der Avatar soll nun aber auditiv, also mit gesprochener Sprache antworten. Ein TTS-System muss die in Textform vorliegende Antwort in Phoneme und schließlich in Phone übersetzen. Über das Internet, einen Audio-Player, Lautsprecher und schließlich die Luft erreichen die Phone, die hörbaren Laute, den Benutzer. Dieser Vorgang ist nun der geschilderten Kommunikationskette zwischen zwei Menschen sehr ähnlich, mit dem Unterschied, dass sich auf der einen Seite kein Mensch sondern eine Maschine befindet, welche die physiologische und linguistische Ebene abdecken muss. Das Gehirn ist durch eine Software ersetzt, die in der Lage ist, Text in Phoneme umzusetzen. Sprachorgane sind durch Software ersetzt, die Phoneme in Phone (hörbare Laute, in diesem Fall auch Audiodateien) umsetzen und diese schließlich über Lautsprecher hörbar machen, also über den Kanal Luft zum Benutzer senden.

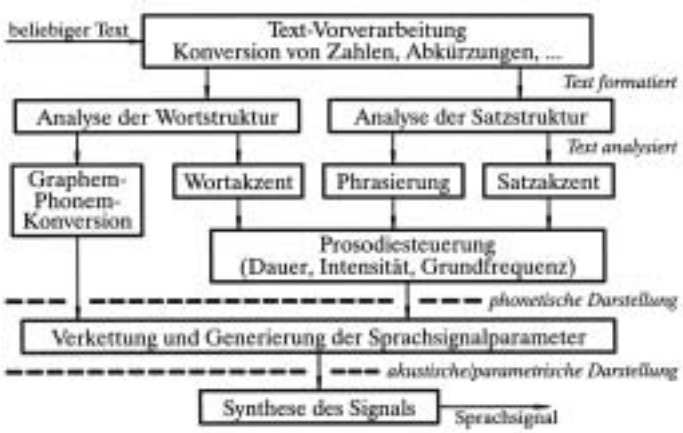

Abbildung 4: Blockdiagramm eines TTS-Systems[9]

Wie realisiert ein Text-To-Speech-System nun die Umsetzung von Text in Phoneme und letztlich Phone? Textgesteuerte Sprachsynthese läuft in drei Schritten ab: Symbolverarbeitung, Verkettung und akustische Synthese. Unter der Symbolverarbeitung sind einige Punkte zusammengefasst, wie die Text-Vorverarbeitung und Analyse der Wort- und Satzstruktur. Hier werden im Wesentlichen Informationen darüber gewonnen, welche Phoneme vorkommen und wie diese später zu verknüpfen sind.

Wortakzent, Phrasierung, Satzakzent oder Prosodiesteuerung[D] müssen unbedingt beachtet werden, da es nicht ausreicht, lediglich Phoneme aneinander zu reihen. Die Tonhöhe steigt beispielsweise gegen Ende einer Frage an, bei einem Ausrufungssatz nicht. Derartige Faktoren müssen Beachtung finden. Je höher entwickelt die entsprechenden Algorithmen sind, desto natürlicher wird sich ein TTS-System anhören. Hierdurch zeichnet sich die Qualität eines solchen Systems aus.

4.2.2. Clientseitiges Text-To-Speech

Seit einigen Jahre werden verschiedene Betriebssysteme mit TTS-Systemen ausgeliefert. Diese Tatsache legt nahe, zu überprüfen, ob sich die Systeme zur Verwendung mit einem Avatar-System eignen. Der Vorteil clientseitiger Systeme ist, dass sie bereits auf dem Computer des Benutzers installiert sind und so keine Audiodaten vom Server zum Client übertragen werden müssen. In diesem Fall werden lediglich die Textdaten übertragen, Audio wird erst auf dem Client erzeugt. Die benötigte Bandbreite ist somit geringer.

Im Folgenden sollen die standardmäßig vorhandenen TTS-Systeme der populären Betriebssysteme Windows und MacOS untersucht werden. Andere clientseitige TTS-Systeme, die nicht standardmäßig in ein Betriebssystem integriert sind, werden hier aufgrund ihrer geringen Verbreitung und des für eine Internetanwendung nicht vertretbaren Installationsaufwands nicht berücksichtigt. Es ist zwar möglich, durch einen automatisierten Installationsprozess auch spezielle TTS-Systeme auf dem Client zu installieren, problematisch sind hierbei jedoch lizenzrechtliche Aspekte, die im Vorfeld mit dem Hersteller des TTS-Systems geklärt werden müssen. Ein weiteres Problem ist die Datenmenge, die bei der Installation eines solchen Systems zum Client übertragen werden muss. Um beispielsweise das Windows-TTS-System mit einer deutschen Stimme zu nutzen, muss

[D] Unter Prosodie versteht die Linguistik die Parameter Quantität, Intensität und Intonation. Diese prägen das Klangbild wesentlich.

für eine Stimme sehr niedriger Qualität mit ca. 0,5 MB Datenaufkommen gerechnet werden, eine Datenmenge, die bei Benutzung über ISDN[E] oder Modem bereits zu Problemen führt. Bei Nutzung der maximalen Bandbreite, die ISDN zu Verfügung stellt, würde der Download bereits 62,5 Sekunden dauern. Diese Bandbreite wird jedoch nur selten erreicht. Bei 50% der maximalen Bandbreite würde der Download also über zwei Minuten beanspruchen. Ein Beispiel für einen Charakter, der unter Windows ein clientseitiges TTS-System verwendet, ist Bubbles[F]. Bubbles ist ein virtueller Fisch, entstanden bei der Kölner Firma Charamel[G]. Um jedoch auch unter MacOS die gleiche Stimme verwenden zu können, wird hier das gleiche TTS-System, das unter Windows clientseitig eingesetzt wird, für MacOS Benutzer nochmals serverseitig verwendet. Eine weitere Schwierigkeit ist also, dass mit clientseitigen Systemen nur schwer auf allen Plattformen die gleiche Stimme eingesetzt werden kann.

Wenn ein clientseitiges TTS-System für einen Avatar einsetzbar ist, so kann es nur ein Standard-TTS sein, das bereits in das jeweilige Betriebssystem integriert ist. Ob dies mit den vorhandenen Systemen unter Windows und MacOS möglich ist, sollen die beiden folgenden Kapitel klären.

4.2.1.1. MacOs Text-To-Speech

Die Firma Apple vertreibt seit dem 17. Oktober 1998[10] mit der Version 8.5 des Betriebssystems MacOS auch das Paket PlainTalk.[11] PlainTalk ist eine Software, die neben englischer Spracherkennung auch ein Text-To-Speech-System für Englisch und mexikanisches Spanisch zur Verfügung stellt. Das System enthält 26 verschiedene Stimmen, vier davon speziell für die spanische Sprache. Die Qualität der erzeugten Sprache ist hochwertig. Produkte von Anbietern, die auf Text-To-Speech Systeme speziali-

[E] ISDN steht für Integrated Services Digital Network, ein Datenübertragungsprinzip, das im Gegensatz zu herkömmlichen Telefonverbindungen mit digitalen Signalen anstelle von analogen Tonfrequenzen arbeitet und eine höhere Übertragungsgeschwindigkeit (64 kbps) erlaubt.
[F] Bubbles ist online im Internet zu sehen: „URL: http://www.bubbleslife.com [Stand: 15.08.2002]“.
[G] Charamel ist eine auf virtuelle Charaktere spezialisierte Firma mit Sitz in Köln (URL: http://www.charamel.com).

siert sind, erzeugen jedoch oft einen natürlicher wirkendes Ergebnis. Diese Systeme sind allerdings meist wesentlich kostenintensiver. Direkt aus einer Webseite auf das PlainTalk Text-To-Speech-System zuzugreifen ist nicht möglich. Hierzu ist es notwendig, ein Plug-in für den jeweiligen Browser zu entwickeln. Apple stellt mit dem Speech Synthesis Manager SDK zwar eine Entwicklungsumgebung für solche Vorhaben zur Verfügung, doch ist die Installation des Plug-ins auf dem Rechner des Benutzers ein großes Hindernis. Hinzu kommt, dass es kaum akzeptabel ist, eine Anwendung im Internet mit einem Plug-in umzusetzen, das lediglich für eine einzige Anwendung dient und ansonsten keinerlei Verbreitung hat. Leider wird die deutsche Sprache von PlainTalk nicht unterstützt. Für die Verwendung im deutschsprachigen Raum kann diese Technologie daher nicht in Frage kommen.

4.2.2.2. Windows Text-To-Speech

Auch Apples Konkurrent Microsoft liefert seit Windows 98, bzw. Windows Windows NT 5.0[12] / Windows 2000 standardmäßig die Microsoft Speech Synthesis Engine aus. Dabei handelt es sich um ein TTS- und Speech-Recognition System. Die Nachteile dieses Systems sind ähnlich, wie die der Apple-Software. So ist die Engine nur über ein ActiveX-Control[H] aus einer Webseite heraus anzusprechen. Zur Entwicklung des ActiveX-Controls stellt Microsoft das Speech Application Programming Interface (SAPI) mit zahlreichen Objekten und Methoden in C/C++ oder Visual Basic zur Verfügung. SAPI ist Microsofts Technologie, um TTS-Funktionalität in Windows-Anwendungen zu integrieren. Weiter steht beim Windows-TTS lediglich eine englische Stimme zur Verfügung, die deutsche Sprache wird nicht unterstützt. Auch die Qualität der erzeugten Sprache entspricht einem ähnlichen Niveau wie bei PlainTalk. Die Probleme des ActiveX-

[H] ActiveX ist eine Browser-Technologie, von Microsoft für den Internet Explorer ab Versionsnummer 3 entwickelt, die es ermöglicht, interaktive Elemente in Webseiten einzubetten. Mit ActiveX-Controls ist es möglich auf Ressourcen des Betriebssystem zuzugreifen und betriebssystemspezifische Befehle abzusetzen.

Controls und der Verwendung des clientseitigen TTS unter Windows sind also prinzipiell dieselben wie bei Apple Macintosh.

4.2.3. Serverseitiges Text-To-Speech

Serverseitige TTS-Systeme bringen zwar den Nachteil mit sich, dass Audiodaten vom Server zum Client transportiert werden müssen, das Datenaufkommen im Vergleich zur Verwendung eines clientseitigen TTS also wesentlich höher ist, doch scheinen die Vorteile zu überwiegen.

Ein wichtiger Punkt, der zwar nicht die technologische Sicht betrifft, jedoch trotzdem nicht außer Acht gelassen werden kann soweit man ein marktfähiges System schaffen will, sind lizenzrechtliche Regelungen. Da die Software, das TTS-System, hier nur einmal auf dem Server installiert ist und nicht auf jedem Client, ist die Nutzung eines dieser Systeme wesentlich günstiger. Einigungen mit den Herstellern sind für eine derartige Verwendung leichter zu erzielen. Die Software ist nur einmal für den Server zu lizenzieren, es müssen keine Regelungen getroffen werden wie mit Lizenzen für Benutzer verfahren wird, auf deren Rechner die Software installiert wird.

Die individuelle Auswahl eines TTS-System, speziell zum Projekt passend, ist ein weiterer Vorteil, der serverseitige Systeme auszeichnet. Jedes System klingt anders, stellt andere Stimmen und Sprachen zur Verfügung und lässt sich anders bedienen. Man hat nun die Möglichkeit, für ein Projekt ein ganz spezielles TTS-System auszuwählen, das den Anforderungen des Projektes, des Charakters am besten gerecht wird. Die Stimme klingt somit auf jedem Rechner gleich, unabhängig vom Betriebssystem. Unsicherheitsfaktoren auf Clientseite, wie z.B. die Sicherstellung der Funktionsfähigkeit eines dort installierten TTS-Systems, entfallen.

Der Markt stellt eine Vielzahl von TTS-Systemen zur Verfügung, die sich zum Einsatz auf einem WWW-Server eignen. Wichtige Kriterien für die

Verwendung mit einem Avatar-System sind vorhandene Schnittstellen, die Möglichkeit eigene Schnittstellen zu definieren und die Qualität der Sprachausgabe. Im Folgenden sollen drei Systeme näher betrachtet werden, die als besonders geeignet erscheinen und später auch bei der Entwicklung zweier Avatar-Systeme Verwendung finden werden.

4.2.3.1. MBROLA

MBROLA ist ein Projekt der Fakultät Polytechnik der Universität Mons, Belgien. Es handelt sich dabei um einen Sprachsynthesizer, welcher Sprache durch das Aneinanderhängen von Phonemen und deren Umwandlung in Phone realisiert. MBROLA ist kein vollständiges TTS-System, da MBROLA normalen Text nicht verarbeiten kann. Vielmehr erwartet MBROLA als Input eine Liste von Phonemen, sowie Informationen über deren Dauer und Tonhöhe.

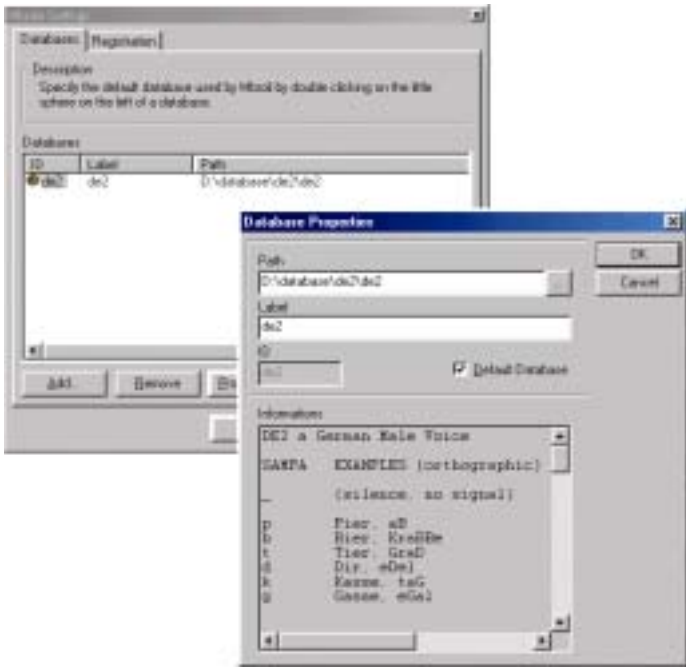

Abbildung 5: MBROLA Control Panel

Das frei verfügbare Software Paket besteht aus dem eigentlichen Sprachsynthesizer und verschiedenen speziell für MBROLA entwickelten Phon-Datenbanken.[13] Eine Phon-Datenbank ist eine Bibliothek von kleinen Abschnitten natürlicher Sprache. Diese Abschnitte werden bei der Synthese aneinander gehängt. Im Control Panel des Sprachsynthesizers können verfügbare Datenbanken angegeben werden, eine Datenbank kann als default definiert werden. MBROLA zeichnet sich dadurch aus, dass eine Vielzahl von Betriebssystemen unterstützt werden. So kann die Software beispielswei-

se unter DOS, Windows, MacOS, Linux, BeOS, Solaris, SunOS oder auch OS/2 installiert werden. Kaum eine TTS-Software ist für derart zahlreiche Betriebssysteme erhältlich und so lässt MBROLA viel Freiraum bei der Auswahl des richtigen Server-Betriebssystems. Auch die Anzahl der verfügbaren Phon-Datenbanken ist bemerkenswert. Über 50 dieser Datenquellen zur Spracherzeugung können mit MBROLA eingesetzt werden. Es handelt sich um weibliche oder männliche Stimmen, unter anderem in den Sprachen Deutsch, Englisch, Spanisch, Italienisch oder Französisch. Selbst exotischere Sprachen wie Kroatisch, Tschechisch, Niederländisch, Griechisch, Hebräisch, Türkisch, Japanisch oder Koreanisch werden angeboten.

Nach dem Modell der lautsprachlichen Kommunikationskette deckt MBROLA lediglich die physiologische Ebene ab. Mit Hilfe der Phon-Datenbank und einer Phonem-Datei erzeugt der Synthesizer Sprache. Die Phonem-Datei enthält die zu verwendenden Phoneme in richtiger Reihenfolge samt Dauer und Tonhöhe. Die Erzeugung dieser Phonem-Datei muss eine Zusatzsoftware übernehmen, die für jede Sprache aufgrund von Sprachunterschieden speziell angepasst sein muss. Aufgabe von MBROLA in einem TTS-System wird demnach die Umsetzung von Phonemen in Sprache, also Phone sein.

Mit dem Kommandozeilen-Programm phoplayer stellt MBROLA ein Werkzeug zur Umwandlung von Phonem-Dateien in Sprache auf Kommandozeilenebene bereit. Phoplayer erzeugt WAV-Dateien[I]. Die Tatsache, das Programm über Kommandozeile und nicht nur über eine grafische Oberfläche bedienen zu können, ist insofern wichtig, als dass damit eine Internetanwendung erstellt werden soll. Ein Kommandozeilenbefehl kann direkt aus einem anderen Skript heraus aufgerufen werden, es kann also eine Schnittstelle zu MBROLAs phoplayer mittels eines weiteren Skriptes erstellt werden. Beispielsweise könnte so ein PHP-Skript später direkt auf

[I] Das WAV-Format ist ein von Microsoft entwickeltes Format für Audiodateien. Wav-Dateien enthalten die rohen Audiodaten, Spuranzahl (mono oder stereo), die Bittiefe, sowie die Samplingrate.

phoplayer zugreifen und durch dynamisch erstellte Phonem-Dateien eine dynamische Spracherzeugung erzielen.

4.2.3.2. HADIFIX

Wie bereits erwähnt, ist MBROLA nicht in der Lage, Phonem-Dateien selbstständig zu erzeugen, es bedarf einer Zusatzsoftware. Mit HADIFIX hat das Institut für Kommunikationsforschung und Phonetik der Universität Bonn ein Sprachsynthesesystem für die deutsche Sprache entwickelt. Diese Software ergibt in Kombination mit MBROLA ein vollwertiges TTS-System.

HADIFIX besteht im Wesentlichen aus dem Programm txt2pho, welches für die Betriebssysteme Windows, Linux und Sun Solaris verfügbar ist. Wie der Name vermuten lässt, erstellt txt2pho aus Textdateien Phonem-Dateien (PHO ist die Dateiendung dieser Dateien). Zuerst wird der Text durch txt2pho in Lautschrift umgesetzt und symbolisch eine Betonung und Phrasierung erzeugt. Als Sprachabschnitte werden Halbsilben und ähnliche Einheiten verwendet, sie umfassen alle Konsonanten vor dem Vokal und den Anfang des Vokals (Anfangshalbsilbe) bzw. das Ende des Vokals und die folgenden Konsonanten (Endhalbsilbe).[14] Das Wort 'Kelch' wird demnach aus 'Ke' und 'elch' gebildet. Zur Erzeugung von Sprache wird aber nicht nur die Lautfolge wiedergegeben, also was gesprochen wird, es muss auch gesteuert werden, wie gesprochen wird. Hierzu gehört die Betonung und die Phrasierung. Im Wesentlichen wirkt die Tonhöhe als Parameter. Bei HADIFIX wird ein Verfahren eingesetzt, so dass bei jeder betonten Silbe die Tonhöhe ansteigt. Dadurch wird die Silbe auch tatsächlich als betont wahrgenommen.[15] Das Wort ‚schnell', das sich inmitten einer Äußerung, also keiner Frage befindet, wird in einer Phonem-Datei beispielsweise durch folgende Zeilen repräsentiert:

```
S 104  12  90  31  92  50  94  69  96  88  98
n 49  16  99  57 100  98 101
E 89  21 102  44 102  66 102  89 101
l 67  15 101  45 101  75 100
```

Eine genauere Erläuterung der Syntax von Phonem-Dateien folgt später, wenn es darum geht, txt2pho an ein Avatar-System anzubinden.

HADIFIX deckt die linguistische Ebene der lautsprachlichen Kommunikationskette ab. Die Software ist fähig, geschriebene Sprache in ihre einzelnen Phoneme aufzutrennen und diese nach festgelegter Syntax in eine Datei zu schreiben. Die Qualität der Sprache, die MBROLA auf Basis der Phonem-Dateien von txt2pho erzeugt, ist hochwertig. Auch kann txt2pho über die Kommandozeile bedient werden, das Programm bietet sich dementsprechend an, in einem Avatar-System Verwendung zu finden.

4.2.3.3. Elan Speech

Elan Speech ist ein etabliertes Unternehmen im Bereich der TTS-Systeme mit Sitz in Toulouse, Frankreich. Die Systeme von Elan Speech decken die Bereiche Telekommunikation, Multimedia, Mobile Systeme und Automobil ab und zeichnen sich durch hohe Qualität und relativ natürliche Sprachwiedergabe aus.

Mit der Software Speech Engine bietet Elan Speech Unterstützung von elf verschiedenen Sprachen bei 20 unterschiedlichen Stimmen. Für die Deutsche Sprache stehen die Stimmen „Dagmar“ und „Thomas“ zur Verfügung. Die Ausgabe erfolgt in Form von WAV-Dateien mit einer Samplingrate von wahlweise 8, 11 oder 16 Khz bei einer Auflösung von 16 bit pro Sample. Mit Elan TTS ist es nicht nur möglich, reinen Text in Sprache zu konvertieren, der Text kann auch sogenannte Text-Tags enthalten. Hierdurch kann die Sprachausgabe beeinflusst werden. Das Tag `\pause{300 ms}` bewirkt beispielsweise eine Sprechpause von 300 Millisekunden. Um Wörter korrekt wiederzugeben, mit denen das System evtl. Probleme hat, gibt es die Möglichkeit, Phoneme direkt anzugeben. `\phone{k."A."p."U."t."sh"^i."n."o."i}` hat als Ausgabe das Wort „Capuccino“. Ein deutsches TTS-System würde die zwei Buchstaben „cc“ in der Regel mit „k“ übersetzen, nicht mit „`t.sh`“. Durch diese und

weitere Tags kann die Sprachausgabe optimiert und genauer akzentuiert werden. Das Ergebnis wirkt naturgetreuer.

Abbildung 6: SAPI TTS-System mit SAPI Manager

Weiter wird die SAPI 4, bzw. SAPI 5.1 Schnittstelle unterstützt.[16] Hierdurch ist es möglich, eine Software zu entwickeln, die über die genannte Schnittstelle auf das Elan TTS-System zugreifen kann. So kann dessen TTS-Funktionalität auch durch ein serverseitiges Skript genutzt werden. Die Software, die per SAPI auf das TTS-System zugreift, soll im folgenden SAPI Manager genannt werden. Der SAPI Manager erhält als Input den zu konvertierenden Text. Daraufhin sucht die Software auf dem Rechner bzw. Server nach einem installierten SAPI TTS-System. Wird dieses gefunden, kann über die SAPI Schnittstelle auf die TTS-Funktionalität zugegriffen werden, der Text wird in eine Audiodatei im WAV-Format konvertiert. Diese Datei erhält der SAPI Manager zurück und kann sie nun entweder direkt an den Client schicken, oder aber vorher in ein entsprechendes Format wandeln und komprimieren. Der SAPI Manager ist eine Windows Anwendung, die in C, C++, Delphi oder Visual Basic zu entwickeln ist.

Prinzipiell kann auf diese Art und Weise jedes beliebige auf dem Server installierte SAPI TTS-System in ein Avatar-System eingebunden werden. Einzige Voraussetzung ist, besagten SAPI Manager zu entwickeln. Es gibt eine Vielzahl von Systemen, welche die von Microsoft entwickelte Schnittstelle SAPI unterstützen. Elan wird daher stellvertretend für andere SAPI TTS-Systeme vorgestellt. Das System ist durch jedes andere SAPI TTS-System austauschbar. Kriterium für die Auswahl eines der Systeme ist daher eher Qualität und Stil der Stimme. Jeder Charakter ist anders und so sollte immer das System gewählt werden, das eine Stimme anbietet, die möglichst gut zum verwendeten Charakter passt. Die Stimme „Thomas“

passt gut zu dem Charakter des Avatar-Systems, wie es im Rahmen dieser Arbeit entwickelt wird. Daher wurde hierfür Elan gewählt.

4.3. Audio-Wiedergabe

Im vorigen Kapitel wurden Technologien zur Erzeugung von Sprache analysiert. Die Sprache liegt nach der Erzeugung in aller Regel in Form von Audiodateien vor, die es auf dem Computer des Benutzers abzuspielen gilt. Meist handelt es sich bei den Dateien um WAV- oder ähnliche Formate, deren Dateigröße ein für die Verwendung im Internet vertretbares Limit deutlich überschreiten. Kompressions- und Streamingverfahren werden zur effizienten Verwendung von Audio im Internet benötigt. Kompressionsverfahren dienen dazu, die Dateigröße der Audiodateien zu verringern. Vorhandene Audioinformation wird effizienter codiert, es werden weniger Daten benötigt. Information, die für den Menschen ohnehin kaum oder gar nicht hörbar ist, wird durch diese Verfahren teilweise entfernt, um weitere Daten einzusparen. Man unterscheidet zwischen verlustfreier und verlustbehafteter Kompression. Da die Datenmenge des Audiomaterials für ein Avatar-System in jedem Falle massiv reduziert werden muss, kommen hier auch verlustbehaftete Algorithmen zum Einsatz. Per Streamingverfahren können Dateien auf dem Client bereits abgespielt werden, bevor sie komplett geladen sind. Die Daten werden quasi an den entsprechenden Player gesendet, wobei eine verbindungsorientierte Übertragung realisiert wird. Die Übertragung erfolgt hierbei nicht über das Protokoll HTTP, sondern über das Real Time Protocol (RTP). Das RTP wurde entwickelt, um Echtzeitverkehr zu unterstützen, d.h. Verkehr, der von der empfangenen Anwendung eine Wiedergabe in einem zeitsensitiven Modus verlangt.[17] RTP stellt somit „Ende-zu-Ende“-Auslieferungsdienste für Daten mit Echtzeitcharakteristika zur Verfügung.

Ein Avatar-System besteht natürlich auch aus einem visuellen Element, dem Charakter, der in einem 3D-Plug-in gerendert wird. Um Ton und Bild, also Sprache und Bewegung zu synchronisieren kann es notwendig sein,

das 3D-Plug-in mittels des Audio-Players zu steuern. Dies ist insbesondere notwendig, falls zur Audioübertragung ein Streamingverfahren zum Einsatz kommt. Synchronisation zwischen Audio und Animation kann dann keinesfalls durch eine fest definierte Timeline erreicht werden. Beim Streaming ist nicht genau vorhersehbar, wann genau welches Wort wiedergegeben wird. Der Stream kann abbrechen oder es muss evtl. zwischengespeichert werden. Das sind Faktoren, die es notwendig machen, die Funktionen zur Synchronisation fest mit dem abzuspielenden Audiomaterial zu verknüpfen. Es muss möglich sein, Schnittstellen zwischen Audio-Player und 3D-Plug-in zu definieren. Dieser Punkt ist deshalb zentrales Thema bei der Analyse der folgenden Technologien.

4.3.1. QuickTime

QuickTime ist ein Dateiformat, eine Reihe von Applikationen und Plug-ins, sowie eine Software-Bibliothek mit einem Application Programming Interface (API) aus dem Hause Apple. Mit dem QuickTime-Plug-in ist es möglich, QuickTime Dateien und noch einige andere Formate innerhalb eines Webbrowsers abzuspielen. QuickTime Dateien werden auch als QuickTime Movies bezeichnet und verwenden die Dateiendung MOV. Der Name Movie sollte jedoch nicht dazu verleiten, diese Dateien mit einem Videoformat, wie z.B. AVI[J] zu verwechseln. QuickTime ist wesentlich vielseitiger. Es kann ein Film sein, jedoch ebenso eine Reihe von Standbildern mit synchronisiertem Text und Hintergrundmusik. *„A QuickTime movie is a file that tells the computer what kind of media to present and when to present it.“* [18] – so eine Definition von Apple.

Das QuickTime Plug-in ist mit 62 Prozent[19] Verbreitungsgrad eine Software, die auf zahlreichen Rechnern vorzufinden ist. Bereits am 10. Oktober 2000 erreichte QuickTime die Marke von 100 Millionen Kopien mit der Programmversion 4.0.[20] Mehr als 100 Millionen Kopien erreichte Version

[J] Technologie der Firma Microsoft, welche die gemeinsame Speicherung von Bild und Ton in einer Datei erlaubt und vor allem in Videosequenzen Anwendung findet.

5.0 gar im ersten Jahr.[21] Die aktuelle Version ist QuickTime 6. QuickTime ist demnach eine weit verbreitete Software, die, soweit sie auch die technischen Anforderungen erfüllt, innerhalb eines Avatar-Systems eingesetzt werden kann.

Ein Element des QuickTime Pakets ist der QuickTime Player, der sich bei Entrichtung der Lizenzgebühr von derzeit $29.99 zu QuickTime Pro updaten lässt. Mit QuickTime Pro ist es möglich, eigene QuickTime Movies zu erstellen. Um die Datenmenge bei diesen Dateien gering zu halten, stellt die Software zahlreiche Codecs zur Kompression zur Verfügung. Für Audiokompression sind das beispielsweise Qdesign Music2[K], MP3[L] oder Qualcomm PureVoice[M]. Die Dateigröße der Audiodateien lässt sich hierdurch erheblich reduzieren. Für die Verwendung von QuickTime Movies im Internet sind die beiden Modi FastStart und Streaming vorgesehen. FastStart ist kein wirkliches Streaming, da es HTTP als Protokoll verwendet, die Datei kann jedoch bereits abgespielt werden, solange sie im Hintergrund noch komplett geladen wird. Daher auch der Name FastStart. Auch wirkliches Streaming ist mit QuickTime-Movies über das Streaming-Protokoll RTP möglich.

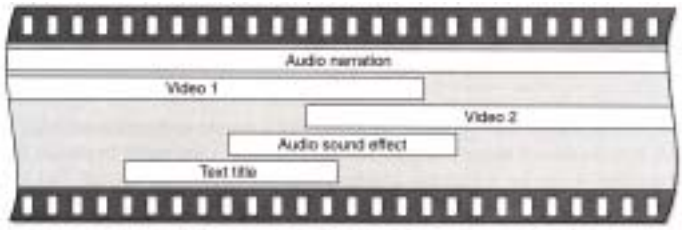

Abbildung 7: Spuren eines QuickTime Movies[22]

Wie schon erwähnt, kann ein QuickTime-Movie in mehreren Spuren organisiert sein. Die Spuren können diverse Elemente, wie Video- oder Audiomaterial, Text, Effekte, etc. enthalten. Ein Movie, wie er für ein Avatar-System zu verwenden wäre, kommt mit lediglich zwei Spu-

[K] Qdesign ist ein Unternehmen aus Kanada, welches einen Codec namens Music2 entwickelt hat. Nach Angaben der Firma ist es damit möglich Audiodaten auf bis zu ein hundertstel der Originalgröße bei guter Qualität zu komprimieren.
[L] Standardformat für komprimierte Audiodateien, das im Rahmen der Moving Picture Experts Group (MPEG) vom Fraunhofer-Institut entwickelt wurde und sich vor allem im Internet verbreitet.
[M] Das kalifornische Unternehmen Qualcomm hat mit PureVoice einen Codec entwickelt, der speziell für die Kompression von Sprache konzipiert ist und so auch im Bereich der Telefonie zum Einsatz kommt.

ren aus. Eine HREF- und eine Tonspur. Die Tonspur enthält die Sprache des Avatars. Wofür wird nun aber die HREF-Spur benötigt? Es kann notwendig sein, die Animationen des Avatars aus dem Audio-Plug-in, in diesem Falle QuickTime, zu steuern. Eine Möglichkeit, eine Schnittstelle hierfür zu entwickeln, bietet die Skriptsprache JavaScript. QuickTime ermöglicht mit den sogenannten HREFTracks, JavaScript-Funktionen an definierten Stellen eines Movies zu rufen. Der Inhalt eines HREFTracks kann mit einem Texteditor erstellt werden und beispielsweise so aussehen:

```
{QTtext}{font:Geneva}{plain}{size:12}{textColor: 65535,
65535, 65535}{backColor: 0, 0, 0}{justify:center}
{timeScale:1000}{width:160}{height:48}
{timeStamps:absolute}{language:0}{textEncoding:0}
[00:00:00.000]
A<javascript:triggerAvatar(,startTalking)> T<_self>
[00:00:02.000]
A<javascript:triggerAvatar(,sayHello’)> T<_self>
[00:00:04.000]
A<javascript:triggerAvatar(,smile’)> T<_self>
[00:00:06.000]
A<javascript:triggerAvatar(,sayBye’)> T<_self>
[00:00:08.000]
A<javascript:triggerAvatar(,stopTalking)> T<_self>
```

Es ist möglich, diese Kommandos in einem HREFTrack abzulegen und das Plug-in im Browser des Clients an den definierten Zeitpunkten somit zu veranlassen, die JavaScript-Funktion `triggerAvatar()` zu rufen. Diese Funktion stellt die Schnittstelle zwischen QuickTime- und 3D-Plug-in dar, beide Plug-ins können so synchronisiert werden.

Da im Voraus jedoch nicht bekannt ist, was der Avatar sagt (die Sprache wird schließlich per TTS erzeugt), muss eine Möglichkeit gefunden werden, die benötigten QuickTime Movies dynamisch zu erzeugen. QuickTime Pro lässt lediglich die Erzeugung mittels einer grafischen Benutzeroberfläche zu. Apple bietet mit dem QuickTime API eine Entwicklungsumgebung, die es ermöglicht, QuickTime-Funktionalität in eigene Applikationen zu integrieren. Das API ist eine Sammlung von C-Funktionen, die un-

ter anderem erlauben, QuickTime Movies zu erzeugen (`CreateMovieFile`), diesen Audio- und HREF-Spuren hinzuzufügen (`NewMovieTrack`) und sie schließlich zu speichern (`ConvertMovieToFile`).[23] Hierdurch ist es möglich, eine eigene Applikation zu entwickeln, die von der Kommandozeile aus gestartet werden kann, als Input ein Audio-File (z.B. WAV) sowie eine Textdatei mit den nötigen Informationen für die HREF-Spur erhält und daraus einen QuickTime Movie erstellt.

QuickTime stellt sich als ausgereifte Technologie zur Präsentation von Mediendaten dar, auch innerhalb eines Webbrowsers. Durch die angesprochenen Fähigkeiten eignet sich QuickTime unbedingt dazu, innerhalb eines Avatar-Systems zur Steuerung eines 3D-Plug-ins und gleichzeitigem Abspielen des Tons eingesetzt zu werden.

4.3.2. RealAudio / Helix

"Seitdem wir vor sieben Jahren RealAudio vorgestellt haben, wartet die Industrie auf eine umfassende, offene Streaming-Plattform. Am heutigen Tag erfüllen wir diesen lang gehegten Wunsch [...]" [24] – Mit diesen Worten kündigte RealNetworks CEO Rob Glaser am 22. Juli 2002 die neue Helix-Plattform an. RealNetworks ist mit dem RealPlayer, dem RealProducer und dem RealServer seit langem ein etablierter Hersteller client- wie serverseitiger Streamingtechnologie. Die letzte Version des clientseitigen Players war RealPlayer 8. Der RealPlayer ist die Software, sowie das Browser-Plug-in, womit RealAudio- und RealVideo-Dateien auf dem Computer angesehen bzw. angehört werden können. Diese Dateien sind mit dem RealProducer zu erstellen, wobei proprietäre Codecs von RealNetworks zum Einsatz kommen. Über den RealServer können die Dateien schließlich per Streamingverfahren zu den Clients übertragen werden. Alternativ gibt es vergleichbar zu QuickTimes FastStart auch hier die Möglichkeit, Real-Dateien per HTTP auszuliefern, diese aber schon während dem Download auf dem Client abzuspielen. RealNetworks hält nach eigenen Angaben einen Marktanteil von 85 Prozent[25] im Bereich Streaming

Media. Eine große Zielgruppe kann demnach durch Verwendung dieser Technologie erreicht werden.

Mit Helix stellt RealNetworks nun sozusagen Version 9 der Produktfamilie vor, wobei einige grundlegende Änderungen eingeführt werden. Helix ist nicht weiter ein proprietäres Produkt der RealNetworks, vielmehr soll die Software geöffnet und der Open-Source Gedanke aufgegriffen werden, um die Software langfristig noch erfolgreicher und verbreiteter zu machen. Erstmals ist RealNetworks bereit, Technologiepartnern und auch der Open-Source-Gemeinde Einblick in den Quellcode der Produkte zu gewähren. Das zwar nicht unter der GNU[N] General Public License (GPL), doch aber unter eigenen, etwas restriktiveren Lizenzen namens RealNetworks Public Source License (RPSL) und RealNetworks Community Source License (RCSL).[27] Hintergedanke von RealNetworks ist, durch das Offenlegen der Technologie und Bereitstellung eines API zahlreiche Partnerfirmen und eine große freie Entwicklergemeinde zu gewinnen, die Anwendungen mit Helix entwickeln und somit automatisch den Marktanteil erhöhen. Mit dem Helix Universal Server bietet RealNetworks zudem einen neuen Streaming Server mit bisher noch nicht realisierten Fähigkeiten. Anbieter von Streaming Content waren bislang mit dem Problem konfrontiert, unterschiedliche Streaming-Server für QuickTime, Real und Windows Media betreiben zu müssen. Mit dem Helix Universal Server existiert nun ein Produkt, das in der Lage ist, alle Forma-

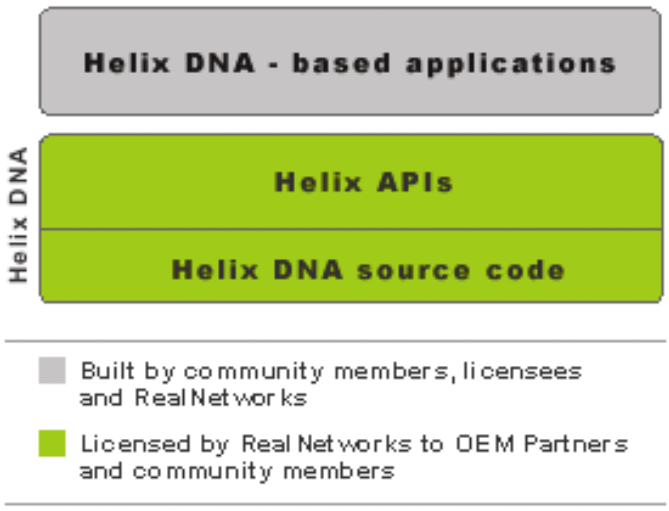

Abbildung 8: Die Helix Plattform[26]

[N] Das GNU Projekt wurde 1984 begonnen, um ein vollständiges Unix-artiges Betriebssystem zu entwickeln, das freie Software ist.

te für unterschiedliche Endgeräte gerecht aufbereitet, von einem Server zu streamen. Es ist davon auszugehen, dass RealNetworks durch die Offenlegung ihres Systems und die Einführung von Helix weiter an Wichtigkeit im Bereich Streaming-Technologie gewinnen und so der Marktanteil des Players weiter steigen wird.

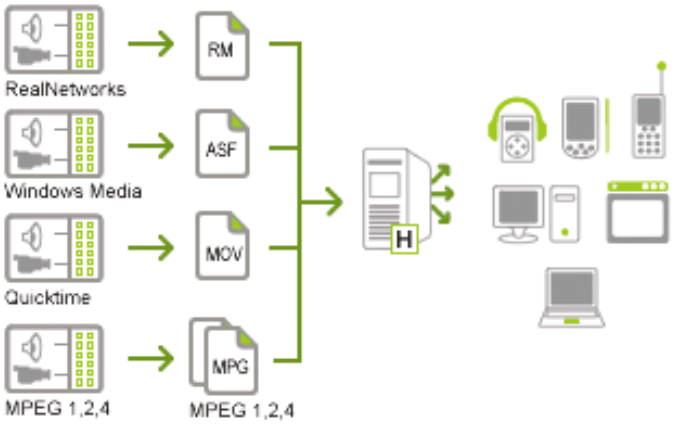

Abbildung 9: Helix Universal Server [28]

Welche Möglichkeiten werden durch RealAudio, bzw. Helix geboten, die für ein Avatar-System von Vorteil sind? Ein Punkt ist sicherlich die gute Kompressionsmöglichkeit für Sprache, die RealAudio zur Verfügung stellt. Der wichtigere Punkt ist jedoch, dass es auch mit Real möglich ist, JavaScript-Funktionen aus einem Audio-Stream heraus aufzurufen und so ein 3D-Plug-in zu synchronisieren. Da Helix erst vor kurzem eingeführt wurde und der entsprechende Player noch nicht in ausreichend großem Maße Verbreitung gefunden hat, werden die entsprechenden Funktionalitäten im Folgenden anhand der Real Version 8 analysiert. Es sei jedoch darauf hingewiesen, dass diese Funktionalitäten auch mit Helix zur Verfügung stehen.

RealProducer Version 8.5 ist die Software, um Content für RealPlayer 8 zu erstellen. Ein großer Vorteil des Producers ist, dass er in einer Version mit grafischer Benutzeroberfläche, wie auch in einer Version für die Kommandozeile ausgeliefert wird. Die Version zur Benutzung per Kommandozeile heißt RMBatch und bietet zusammen mit den beiden Tools RMEditor und RMEvents den gleichen Funktionsumfang wie das Pendant mit GUI[O].

[O] GUI bedeutet Graphical User Interface. Vorzufinden bei Software, die das Benutzen eines Systems oder einer Applikation durch den Einsatz von Mausklicktechnik, Icons und Scroll-Balken komfortabel macht.

RMBatch ist zum Erzeugen von RealMedia-Dateien (Dateiendung RM) einzusetzen. Mit RMEditor können die Dateien nachträglich bearbeitet werden und RMEvents lässt die Einbindung einer Skriptdatei in solch eine Datei zu. In einer Skriptdatei können Events festgelegt werden, die beim Abspielen der Datei an definierten Zeitpunkten ausgelöst werden. Ein Event ist das URL-Event, mit dem es auch möglich ist, JavaScript-Funktionen zu rufen. Die Syntax innerhalb einer Skriptdatei muss dem Schema `u <starttime> <endtime> <URL>` entsprechen[29], ein Beispiel könnte so aussehen:

```
u 00:00:00.000 00:00:01.000
javascript:triggerAvatar(,startTalking)
u 00:00:02.000 00:00:03.000
javascript:triggerAvatar(,sayHello’)
u 00:00:04.000 00:00:05.000
javascript:triggerAvatar(,smile’)
u 00:00:06.000 00:00:08.000
javascript:triggerAvatar(,sayBye’)
u 00:00:08.000 00:00:10.000
javascript:triggerAvatar(,stopTalking)
```

Wie bei QuickTime können also auch hier Zeitstempel definiert werden, um Events auszulösen, welche wiederum JavaScript-Funktionen rufen können. Mit dem Befehl `rmevents -i input.rm -e events.txt -o output.rm` kann die Skriptdatei events.txt der Real-Datei input.rm hinzugefügt werden. Als Output wird das File output.rm erzeugt, welches im Falle eines Avatar-Systems Audio-Information wie auch Skriptbefehle enthalten würde.

Wie QuickTime macht auch die Technologie von RealNetworks einen guten Eindruck was deren Verwendung in einem Avatar-System angeht. Von Vorteil bei der Verwendung von Real ist, dass bereits Kommandozeilenprogramme vorliegen, mit denen Real-Dateien auch ohne Benutzeroberfläche auf Skriptebene erstellt werden können. Die Möglichkeit, eine JavaScript-Schnittstelle über Skriptdateien an das Real-Plug-in im Browser anzubinden, ist ebenfalls gegeben.

4.3.3. Windows Media

Advanced Streaming Format, kurz ASF – so bezeichnet die Firma Microsoft das hauseigene Format für Streaming-Media. ASF-Dateien werden auf dem Computer durch den Windows Media Player wiedergegeben. Der Player ist Bestandteil der Windows Media Produktfamilie, die mittlerweile in der Version 8 vorliegt. Bereits Mitte des Jahres 1997 führte Microsoft mit NetShow 2.0 den Vorgänger von Windows Media ein.[30] NetShow ist ein Client-/Server-System, das Audio, Video oder Slideshows live, bzw. on-demand streamt oder per HTTP ausliefert. Netshow legte für Windows Media den Grundstock und beinhaltet einige Elemente, die so in folgenden Versionen übernommen wurden. Komponenten sind der NetShow Player, NetShow Server und die NetShow Authoring Tools. NetShow Player ist Vorgänger des Windows Media Player, das clientseitige Programm zum Abspielen von ASF-Dateien. Als ActiveX Komponente ist der Player bereits seit Version 4.0 mit dem Internet Explorer kombiniert, welcher wiederum fester Bestandteil des Betriebssystem Windows ist. NetShow Server ist Vorgänger des Media Servers, ein Streaming Server, welcher standardmäßig als Windows Media Service unter Windows 2000 verfügbar ist. Die NetShow Authoring Tools sind Programme, die zum Erstellen und Bearbeiten von Windows Media Content benötigt werden. Die Programme wurden kontinuierlich weiterentwickelt und sind so auch Bestandteil der Nachfolgeversionen Windows Media Encoder 7.1 Tools und Windows Media Tools 4.1. Es handelt sich dabei um eine breite Softwarepalette, die von Kommandozeilenprogrammen bis hin zu vollwertigen Windows Applikationen mit grafischer Oberfläche reicht. Alle Komponenten der Windows Media Produktfamilie sind frei erhältlich, lediglich für den Streaming Server wird Windows 2000 als Betriebssystem benötigt.

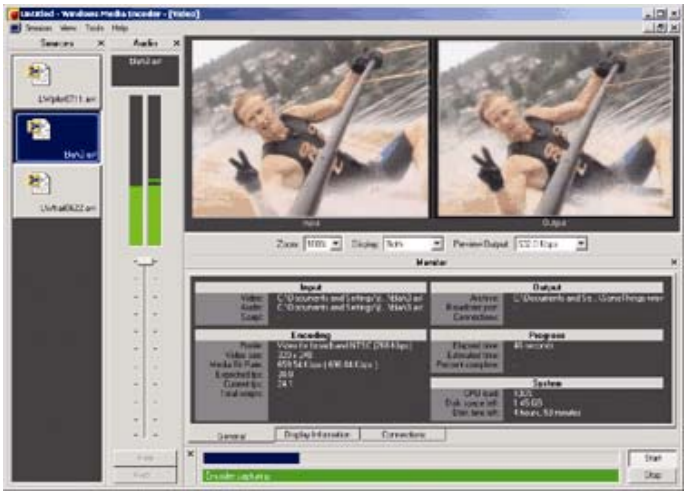

Abbildung 10: Windows Media Encoder 7.1[31]

Ein Werkzeug zum Erstellen von Windows Media Content ist der Windows Media Encoder, der aktuell in der Version 7.1 vorliegt. Windows Media Encoder erlaubt die Erstellung von ASF-Dateien unter Verwendung verschiedenster Kompressionsalgorithmen. Über eine grafische Oberfläche kann eine Vielzahl von Formaten in ASF konvertiert werden, darunter z.B. WAV, AVI oder MP3. Mit dem Programm wav2asf, Element der Windows Media Tools 4.1, existiert außerdem ein Kommandozeilenprogramm, womit WAV-Dateien auch ohne grafische Oberfläche in ASF konvertiert werden können. Das Programm kann über die DOS-Shell, wie auch direkt über ein Skript nach folgender Syntax aufgerufen werden:

```
wav2asf –in c:\soundfiles\wav\sound.wav –out
c:\soundfiles\asf\sound.asf
```

Bereits seit Netshow 2.0 bietet auch die Microsoft Technologie die Möglichkeit, Skript-Kommandos in eine ASF-Datei zu integrieren. Auf diese Weise kann ein NetShow, bzw. Windows Media Player ActiveX-Control aus einer Webseite heraus JScript[P] Funkionen aufrufen und so ein 3D-Plug-in synchronisieren. Das Kommandozeilenprogramm hierfür heißt ASFChop und erwartet als Parameter den Namen der Input-Datei, den Namen der Output-Datei, die erzeugt werden soll, und den Namen einer Skriptdatei, welche die Kommandos enthält. Der Aufruf des Programms muss nach folgender Syntax erfolgen.

[P] JScript ist das Pendant der Firma Microsoft zu JavaScript. JScript bietet zusätzlich zum Funktionsumfang von JavaScript einige spezielle Funktionalitäten, die ausschließlich im Microsoft Browser, dem Internet Explorer zum Einsatz kommen können.

```
ASFChop -in c:\soundfiles\asf\sound.asf
-out c:\soundfiles\asf_index\sound_index.asf
-script c:\soundfiles\scripts\sound_index.txt
```

Die zu verwendende Skriptdatei ist im Textformat zu erstellen und kann entweder durch einen normalen Texteditor oder aber dynamisch durch ein Skript erzeugt werden. Eine Skriptdatei zur Erzeugung von Skriptkommandos in einer ASF-Datei hat folgenden Aufbau:

```
start_script_table
00:00:00.0 startTalking XParamX
00:00:02.0 sayHello XparamX
00:00:04.0 smile XparamX
00:00:06.0 sayBye XparamX
00:00:08.0 stopTalking XparamX
end_script_table
```

Zu den definierten Zeitpunkten löst das im Browser des Clients befindliche ActiveX Control ein Event names `ScriptCommand` aus, welchem die in der Skriptdatei festgelegten Parameter übergeben werden.[32] Durch JScript kann dieses Event wiederum abgefangen werden und entsprechende Aktionen ausführen. Folgender JScript-Code ist hierzu in die HTML-Seite zu integrieren:

```
<SCRIPT FOR="MediaPlayer"
EVENT="ScriptCommand(bstrType, bstrParam)"
LANGUAGE="Jscript">
    3dplugin.trigger(bstrType.toLowerCase());
</SCRIPT>
```

In diesem Falle würde der Eventhandler `ScriptCommand` die Methode `trigger()` des 3D-Plug-ins rufen und dieser einen Parameter senden (z.B. `startTalking`, `sayHello`, etc.). Die Methode `trigger()` kann den 3D-Charakter dann wiederum veranlassen, entsprechend zu reagieren, Synchronisation zwischen Audio und Animation kann somit erreicht werden. Auch die Windows Media Technologie eignet sich somit dazu, in einem Avatar-System eingesetzt zu werden.

Abbildung 11: Verbreitungsstatistik Plug-ins[33]

Da der Windows Media Player fester Bestandteil von Windows ist, ist dessen Verbreitung in den letzten Jahren rapide gestiegen. Web-Hits stellt eine verbreitung von 80,2 Prozent fest. Zu den Verbreitungszahlen der drei Plug-ins QuickTime, Real/Helix und Windows Media Player ist jedoch zu sagen, dass die von unterschiedlichen Quellen veröffentlichten Zahlen stark variieren und so nur bedingt ein Kriterium zur Festlegung auf eine dieser Technologien sein können. Auch können sich diese Daten binnen relativ kurzer Zeit massiv ändern. Grund dafür sind eventuelle technologische Vorsprünge, die ein Hersteller erzielen kann oder aber strategische Entscheidungen und Partnerschaften, welche die Verbreitungszahlen verändern. So ist klar festzustellen, dass Microsofts Entscheidung, den Windows Media Player in Windows zu integrieren, dazu geführt hat, den Marktanteil drastisch zu erhöhen. Ob Entscheidungen wie z.B. die Offenlegung des Real/Helix-Systems zu ähnlichen Entwicklungen führen, bleibt abzuwarten.

Es ist festzustellen, dass noch nicht klar ist, welche der drei Technologien sich in Zukunft zum Marktführer etablieren wird und so einen Standard setzen kann. José Alvear schrieb zwar bereits vor vier Jahren *„If all goes according to Microsoft's plan, NetShow will be the market leader in streaming multimedia, an their ASF Files will be ubiquitous so that ASF becomes the standard streaming file format in the Internet“* [34], das Rennen um die Marktführung scheint jedoch bisher noch von keinem der drei Anbieter gewonnen zu sein. Ein Avatar-System sollte daher von vornherein so geplant sein, dass, falls eine der Technologien eingesetzt wird, sichergestellt ist, alternativ auch eine der beiden anderen einsetzen zu können. Eventuell kann ein System sogar alle drei Varianten anbieten und dem Benutzer den Content liefern für den er das passende Plug-in installiert hat. Der re-

lativ ähnliche Leistungsumfang der drei Technologien, bezogen auf die Anforderungen eines Avatar-Systems, sollte dies ermöglichen.

4.4. 3D-Technologien

3D-Technologien versuchen bereits seit einigen Jahren im Internet Fuß zu fassen. Den Anstoß hierzu gab die sogenannte Virtual Reality Modeling Language (VRML). VRML erlebte seine Geburtsstunde im Frühjahr 1994 auf der ersten World Wide Web Konferenz in Genf, wo Marc Pesce und Anthony Parisi ein Konzept für eine dreidimensionale Benutzerschnittstelle für das World Wide Web vorstellten. Es wurde angeregt, eine Beschreibungssprache zu entwickeln, mit der sich dreidimensionale Welten definieren lassen, wobei auch die Integration von Hyperlinks möglich sein sollte.[35] Daraufhin wurde bereits 1995 die VRML Version 1.0 der Öffentlichkeit vorgestellt. 1996, also nur ein Jahr später, wurde auch schon Version 2.0 veröffentlicht, die auf Basis des Open Inventor ASCII File Formats der Firma Silicon Graphics[Q] entwickelt wurde. *„Dieses Format unterstützt die vollständige Beschreibung von dreidimensionalen Szenarien mit Objekten, Lichtquellen, Oberflächentexturen und realistischen Effekten“* [36] - es bot sich also an, es an die Bedürfnisse von VRML anzupassen und als Basis zu verwenden. Am 22.Juli 2002 hat das Web3D Consortium mit Extensible 3D (X3D) die aktuelle Version von VRML verabschiedet.[37]

[Q] Silicon Graphics Inc. oder auch SGI bezeichnet sich selbst als Marktführer in den Bereichen Hochleistungsrechner, Visualisierungen und komplexen Datenmanagements. Die Firma hat ihren Sitz in Mountain View, Kalifornien und besteht bereits seit 20 Jahren.

Abbildung 12: VRML Spiel: Second Nature World[38]

VRML, bzw. X3D-Szenen können im Browser durch ein entsprechendes Plugin angezeigt werden. Die Information wird dabei aus Dateien mit der Endung WRL gelesen, die Szene daraufhin durch das Plugin gerendert. Die Grundlage jeder Szene wird rechnerintern durch eine Menge von Punkten im Raum gebildet. Mit Hilfe dieser Punkte werden sichtbare geometrische Objekte konstruiert. Die Oberfläche dieser Objekte wird dadurch gebildet, dass auf der Basis der Punkte Mengen von ebenen Vielecken (auch Polygone genannt) definiert werden, die mit ihren Flächen die Außenhaut bilden.[39] Bei der Verwendung im Internet ist darauf zu achten, die Polygonzahl möglichst gering zu halten. Je mehr Polygone, desto rechenintensiver ist die Darstellung der Szene und desto größer die zu übertragende Datenmenge. VRML / X3D Szenen, wie man sie im Internet findet, haben von daher häufig eine recht kantige Erscheinung. Meist sind es Spiele, virtuelle Räume, wie z.B. Galerien oder Produktvisualisierungen, die man im Internet im VRML-Format sieht.

„Durch die Integration in die beiden am meisten verbreiteten WWW-Browser wird VRML 2.0 im Internet bald allgegenwärtig sein.“ [40] – so prognostizierte es Hans-Lothar Hase 1997. Die Entwicklung konnte in dieser Form allerdings nicht beobachtet werden und so sind VRML-Anwendungen auch heute noch eine eher exotische Erscheinung im Internet. Es mag an der unvollkommenen Erscheinung der Szenen, der hohen Beanspruchung der Rechnerleistung oder aber am verhältnismäßig umfangreichen Datenaufkommen bei oft langsamen Netzverbindungen liegen. Tatsache ist, dass VRML trotz der offiziellen Anerkennung als WWW-Standard durch das W3 Consortium der große Durchbruch bis heute nicht

gelungen ist. So drängen mittlerweile zahlreiche proprietäre Produkte auf den Markt, um in diesem Bereich Fuß zu fassen und sich zu etablieren. Zwei dieser Technologien sind die Viewpoint Experience Technology (VET) und CharActor. Da diese aufgrund spezieller Eigenheiten zur Verwendung innerhalb eines Avatar-Systems besonders geeignet scheinen, sollen sie daraufhin im Folgenden genauer untersucht werden.

4.4.1. Viewpoint Experience Technology

Die Viewpoint Experience Technology (VET) ist eine Entwicklung der Viewpoint Corperation. Es handelt sich dabei um eine Technologie, mit der es möglich ist, Rich Media Content progressiv über das Internet zu laden und schließlich im sogenannten Viewpoint Media Player (VMP) darzustellen. VET ermöglicht die Kombination vielerlei Arten von Content, einschließlich 2D Grafiken, 3D Modellen, Animationen, Macromedia Flash Filmen, Audio und Text. Der VMP ist für die gängigen Browser Netscape und Internet Explorer für die Plattformen Windows wie auch MacOS frei erhältlich. Die 3D-Darstellung des Players ist sehr hochwertig, 3D Content wird durch das Plug-in direkt im Browser des Clients gerendert. Die hohe Qualität ist durch das verwendete Verfahren begründet. Normalerweise werden 3D-Szenen Bild für Bild errechnet und dargestellt. Die Basis für Viewpoint Szenen bilden jedoch nicht Bilder, sondern die Zeit.[41] Eine Animation wird daher nicht beschrieben durch den Ausdruck „bewege den Ball von A nach B in 15 Bildern“, sondern vielmehr durch „bewege den Ball von A nach B in 1 Sekunde“. Ein langsamer Rechner, der nur 10 Bilder pro Sekunde rechnen kann, würde die Animation dann in 10 Bildern darstellen, ein schnellerer vielleicht in 15 Bildern. Der Effekt ist, dass die Animation überall die gleiche Dauer hat und auf jedem Rechner das Maximum der möglichen Darstellungsqualität erzielt wird.

Die Basis einer Viewpoint Szene bilden zwei Dateien, die MTS- und die MTX-Datei. Für die populären 3D Programme 3d studio max, Cinema 4D, Lightwave 6.5 und Maya gibt es die Möglichkeit, Szenen durch ein speziel-

les Plug-in zu exportieren und so MTS- und MTX-Dateien zu erstellen.[42] Eine MTS-Datei ist eine komprimierte Binärdatei, welche die einzelnen Szenenelemente enthält. Diese Elemente werden auch Media Atoms genannt. Es handelt sich dabei um progressiv ladbare Grundelemente wie 3D-Objekte, Sounds, Movies oder Flash-Animationen. Ergänzen lassen sie sich durch Procedural Media Atoms, das heißt Standardelemente, wie Lichtquellen, Grundfiguren oder Texte, die bereits im Player-Plug-in enthalten und dadurch sofort verfügbar sind.[43] Die MTX-Datei ist eine XML-basierte Datei, welche die hierarchischen Beziehungen zwischen den einzelnen Szenenelementen enthält und diese innerhalb der Szene platziert. Es handelt sich um eine editierbare Datei, man kann durch Veränderung des XML-Codes Effekte erzielen oder Animationen programmieren. Die MTX-Datei ist sozusagen die Kommandozentrale einer Viewpoint Szene. Wichtig für das Verständnis ist, dass Viewpoint XML nicht in herkömmlicher Weise verwendet. Ein gängiger Einsatz von XML ist die Beschreibung strukturierter Daten, z.B. für eine Webseite. Viewpoint benutzt die XML-Struktur hingegen um 3D und Rich Media Szenen zu beschreiben, als eine Sprache um die Szene für den VMP lesbar zu machen und die hierarchischen Beziehungen der Szenenelemente zu koordinieren.

XML auch zur Strukturierung von 3D Daten zu verwenden liegt eigentlich nahe. XML ermöglicht es, ein Dokument genau zu beschreiben, und zwar so, dass ein Rechner es „verstehen“ kann. Auch eine 3D Szene kann als ein solches Dokument verstanden werden. Sie besteht ebenfalls aus vielen Elementen und Parametern, die in Relation zueinander stehen. XML ist so aufgebaut, dass jedes bedeutungsvolle Element einer kohärenten Baumstruktur identifiziert werden kann, mit der sowohl ein Mensch als auch ein Computer arbeiten kann.[44] Ein Auto kann beispielsweise als vollständige Liste seiner Teile beschrieben werden, alles, vom Motor bis zu den Grundbestandteilen, wird dann auf eine Liste der Komponenten zurückgeführt. Bei einem Avatar, der mit Viewpoint erstellt ist, würden derartige Komponenten durch Media Atoms repräsentiert. Durch XML sind diese in Beziehung zueinander zu setzen. So gibt es den Avatar, der untergeordnete Elemente wie z.B. Hand oder Finger hat.

```
<avatar>
    <hand>
        <finger parameter1=“”>
        <finger parameter1=“”>
        <finger parameter1=“”>
    </hand>
</avatar>
```

Dieses Codebeispiel ist nun sehr simpel und soll lediglich die Struktur verdeutlichen, wie ein 3D-Objekt durch XML zu gliedern ist. Wie die Struktur genau auszusehen hat, muss durch eine sogenannte Document Type Definition (DTD) festgelegt werden. XML stellt mit Hilfe der DTD einen Mechanismus zur Verfügung, mit dessen Hilfe Wissen über die Struktur der Daten zugänglich gemacht werden kann.[45] Im Fall Viewpoint definiert die Viewpoint Corporation, wie die Datenstruktur auszusehen hat. Die DTD ist in das Player-Plug-in eingearbeitet, welches sie zum Lesen und Verarbeiten der XML-basierten MTX-Dateien braucht.

Die Struktur einer MTX-Datei ist komplex. Eine Szene wird durch vier Basissektionen beschrieben. MTSScene-Params enthält allgemeine Angaben und Parameter zum Render- und Beleuchtungsmodell oder den Antialiasing Eigenschaften. MTSCameraMode steuert die Einstellungen der virtuellen Kamera durch die der Benutzer die Szene betrachtet. Durch

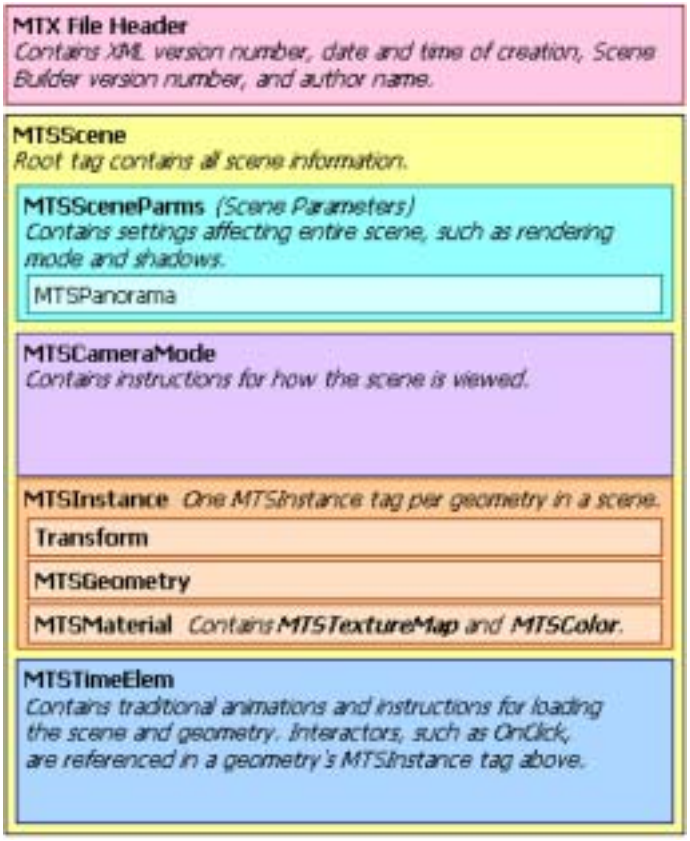

Abbildung 13: MTX-Datei: XML Struktur[46]

MTSInstance werden die hierarchischen Beziehungen der einzelnen Media Atoms geregelt. Dieser Bereich ist sehr umfangreich, da alle Elemente richtig in Verbindung zueinander gebracht werden müssen.

```
<MTSInstance Name="rFoot_1" Visible="0" DoShadow="0" >
    <MTSInstance Name="rToe_1" Visible="0" DoShadow="0"
    </MTSInstance>
</MTSInstance>
```

Der Zeh muss beispielsweise Unterelement des Fußes sein. Auf diese Art und Weise ist die gesamte dreidimensionale Erscheinung zu gliedern. MTSTimeElem enthält Timelines und sogenannte Interactors. Über Timelines können Animationen realisiert werden. Dabei werden zu definierten Zeitpunkten Eigenschaften der mit MTSInstance platzierten Elemente geändert. Die Animation eines Elementes kann z.B. durch kontinuierliche Veränderung der Positions- und Skalierungswerte realisiert werden. Zusätzlich können per MTSTimeElem auch externe Elemente, wie z.B. MP3-Audio, eingebunden werden.

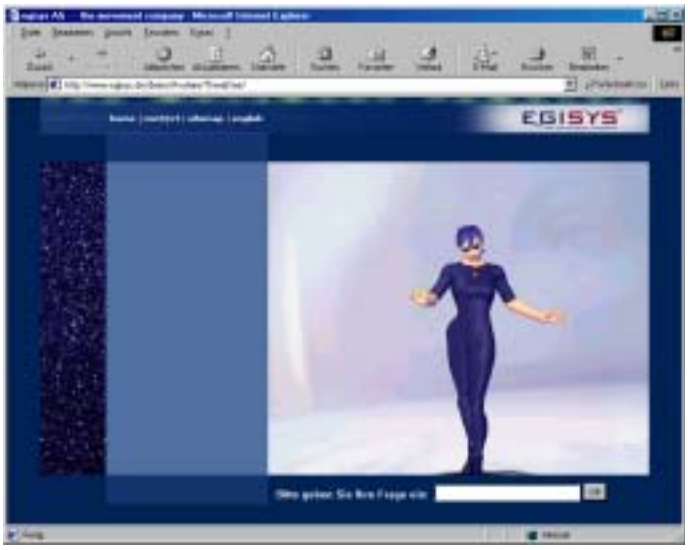

Abbildung 14: Viewpoint Charakter: ThreeDee[R]

Es gibt bereits Avatare, die VET verwenden. Ein Beispiel ist der Avatar ThreeDee der Firma Egisys. Dieser ist jedoch so konzipiert, dass er zur Audioausgabe vorgefertigte MP3-Dateien verwendet, dynamische Spracherzeugung ist nicht vorgesehen. Genau hier liegt die Problematik der VET. Mit statischen Audiodateien lassen sich Animationen genau passend zum Ton erstellen, diese als Viewpoint

[R] ThreeDee ist online im Internet zu sehen: “URL: http://www.egisys.de/demo/Avatare/ThreeDee/ [Stand: 28.08.2002]”.

Dateien exportieren und schließlich in das Avatar-System integrieren. Ist die Audioausgabe dynamisch, so lassen sich keine passenden Animationen vorproduzieren. *„Zu den schwierigsten Aufgaben für 3D-Designer zählen lippensynchrone [...] Gesichtsanimationen“* [47], das weiß auch Jörn Laviscach vom Heise Verlag. Sollen die Lippenbewegungen dynamisch erzeugt werden, wird es noch um ein Vielfaches schwieriger.

Optimal wäre für jedes Phonem eine Mundstellung zu definieren, zwischen denen, je nach auszugebendem Laut, gemorpht wird. Unter Morphing versteht man im 3D-Bereich die Angleichung einer Oberflächenform an eine andere innerhalb eines definierten Zeitraumes. Wird vom Phonem „O“ zum Phonem „Sch“ gemorpht, so wird die Oberflächenform also angeglichen, der Mund schließt sich ein wenig und die Mundöffnung hat nicht mehr eine ganz so runde Fom. Dies ist mit VET jedoch nicht zu realisieren. Es ist zwar möglich, verschiedene Morphtargets (das sind die definierten Zustände zwischen denen gemorpht werden kann) zu definieren, Morphing zwischen diesen ist jedoch nur über eine feste Timeline möglich. Diese Tatsache bringt für die Synchronisation zwischen Ton und Bild viele Probleme mit sich.

Eine Möglichkeit TTS-Ton einzubinden ist die dynamische Erstellung einer MP3-Datei, welche die Sprache enthält und die Generierung einer MTX-Datei, welche die entsprechende Timeline mit den Morphanweisungen enthält. Die MTX-Datei muss dann in den VMP geladen werden. In dieser Datei ist auch die MP3-Datei referenziert. Die Timeline triggert schließlich die Morphing Animationen, gleichzeitig spielt der Player das MP3 ab. Problematisch ist hierbei, dass Audiowiedergabe und Lippenbewegungen nicht direkt aneinander gekoppelt sind. Würde der Audiostream abbrechen, so würden die Lippenbewegungen aufgrund der vorhandenen Timeline weiterlaufen. Wirkliche Synchronisation kann also nicht erreicht werden.

Eine zweite Möglichkeit wäre das Morphing der Mundpartie von einem externen Audioplayer über Skriptkommandos zu steuern. Als Schnittstelle

zwischen Audioplayer und VMP kommt auf Clientseite allerdings nur JavaScript in Frage. Der Zugriff per JavaScript auf ein Browser-Plug-in ist ein Vorgang, welcher der JavaScript Engine viele Ressourcen abfordert. Offizielle Zahlen sind hierzu nicht zu finden. Ein selbst durchgeführter Software-Test hat allerdings ergeben, dass per JavaScript abgesetzte Befehle schon bei einer Häufigkeit von mehr als einem Befehl pro Sekunde von JavaScript und dem Viewpoint Plug-in nicht mehr sauber verarbeitet werden.[S] Es ist also davon auszugehen, dass die entsprechenden Befehle zum Steuern der Mundpartie nicht in der nötigen Geschwindigkeit transportiert bzw. ausgeführt werden können. Bei den Phonemen innerhalb der Sprache ist im Millisekundenbereich zu rechnen, also wären die Ausführungszeiten für dieses Modell zu kurz.

Bei nicht real wirkenden Charakteren, also z.B. Fantasie- oder Comicfiguren, bietet sich außerdem die Möglichkeit, eine Standardanimation für die Rede zu entwickeln. Diese Animation beginnt und endet mit der gleichen Position und Stellung des Mundes und wird solange wiederholt, wie Audiomaterial zur Verfügung steht. Fängt der Avatar an zu reden, muss die Animation gestartet werden, hört er auf, muss man sie stoppen. Hierfür sind lediglich zwei Skriptkommandos notwendig, das kann die JavaScript-Schnittstelle bewältigen.

Trotz der Probleme, dynamisch erzeugten Ton in ein Avatar-System mit VET zu integrieren, scheint die Technologie durchaus einsetzbar für ein solches System zu sein. Gerade für Kunstfiguren, von denen ohnehin keine exakten menschlichen Bewegungen erwartet werden, bietet Viewpoint die Möglichkeit, eine sehr hohe Qualität des 3D-Renderings zu erzielen. Ein weiterer Vorteil ist, dass die Software momentan immer populärer wird und so mit einer raschen Verbreitung des Plug-ins zu rechnen ist. *„Im kommenden AOL-Client 7 [...] ist View Points 3D-Technologie bereits enthalten“* [48] – so die Feststellung von Frank Puscher, Redakteur der Internet

[S] Getestet unter Windows 98, Pentium II, 333Mhz, 192 MB RAM. Über JavaScript Befehle wurde vom Windows Media Player 7 auf das Viewpoint Plug-in zugegriffen. Beide Plug-ins befanden sich innerhalb des Internet Explorers 5.5.

World. Tatsächlich ist es so, dass die Verbreitung des Plug-ins über AOL dem Produkt gute Chancen bietet, sich im Markt zu etablieren. AOL ist dem aktuellen TV-Werbespot zufolge Marktführer mit 35 Millionen Kunden weltweit.

4.4.2. CharActor

CharActor ist eine bislang noch relativ unbekannte 3D-Lösung der Firma Charamel aus Köln. Es handelt sich um eine Internet-Software, welche die Platzierung interaktiver dreidimensionaler Charaktere im Internet ermöglicht. Die Verbreitung des Plug-ins ist noch gering. Da die Technologie jedoch speziell für den Einsatz innerhalb eines Avatar-Systems enorme Vorteile und einen großen technologischen Vorsprung aufweist, soll an dieser Stelle genauer auf Funktionsweise und Leistungsfähigkeit eingegangen werden.

Die Firma Charamel hat die CharActor-Technologie speziell dafür entwickelt, interaktive 3D Charaktere im Internet und anderen interaktiven Medien einzusetzen. Ein Charakter für das Plug-in liegt daher als dreidimensionales Modell vor. Die Oberfläche oder auch die Haut liegt in Form eines Gitternetzes vor, welches die äußerliche Form beschreibt. Weiter besitzt der Charakter ein Skelett, das bei einem menschlichen Charakter beispielsweise dem menschlichen Skelett nachgebildet ist. Über das Skelett lassen sich die einzelnen Glieder des Charakters bewegen, so können die Körperbewegungen gesteuert werden.[49]

Zur Bewegungen des Charakters liegen Bewegungsdateien vor (Dateiendung CCM). Diese Dateien enthalten Informationen darüber, welches Glied des Skelettes zu welchem Zeitpunkt wie transformiert werden soll. Bewegungsdateien können im Internet käuflich erworben werden. Diese Dateien eignen sich aber für speziell entwickelte Fantasiecharaktere nur selten, da die Bewegungen nicht an die spezifische Erscheinung angepasst sind. Alternativ gibt es die Möglichkeiten, Bewegungsdateien per

Motion-Capture Verfahren oder durch Animation in einem 3D-Programm zu erstellen. Beim Motion-Capture Verfahren werden die Bewegungen eines Schauspielers verfolgt (z.B. durch Kameras), registriert und in Animationssequenzen übersetzt. Die erfassten Bewegungsdaten können dann auf das Skelett des Charakters angewendet werden. Auch dieses Verfahren führt bei der Verwendung von Fantasiefiguren jedoch oft zu Problemen. Eine Comic-Figur erzielt häufig gerade durch ihre nicht menschlichen Bewegungsabläufe die erwünschte Wirkung. Ein Schauspieler muss diese Bewegungsabläufe also möglichst gut produzieren, um Ergebnisse zu erzielen, die mit wenig Nachbearbeitung übernommen werden können. Bei der Animation von Hand in einem 3D Programm können die Bewegungsabläufe hingegen sehr exakt und genau erstellt werden. Nachteil ist die oft lange Produktionszeit dieser manuell erstellten Animationen. Zum Export der Animationen, wie auch des Charakters selbst, hat Charamel ein Plug-in für die Software 3d studio max entwickelt.

Um einen Charakter mit CharActor auf einer Webseite einzusetzen, sind verschiedene Dateien auf einem Webserver zugänglich zu machen. Da gibt es z.B. die Installationsdateien des Plug-ins für verschiedene Browser und Betriebssysteme. Aktuell ist das Plugin unter Windows für Internet Explorer und Netscape ab Version 4 verfügbar, unter MacOS wird Netscape ab Version 4 unterstützt. Nach Angaben von Charamel sollen auch Plug-ins für Netscape 6 auf Windows und MacOS in Kürze fertiggestellt sein. Das Modell des Charakters liegt als Datei mit der Endung CIM vor. Behave.cfg ist eine Datei mit Angaben zur Steuerung lippensynchroner Mundbewegungen. Bei Verwendung eines TTS-Systems enthält sie zudem Informationen zur Stimme wie Geschlecht, Tonhöhe und Schnelligkeit.[50] Über eine Verzeichnishierarchie wird eine Art Bewegungsbibliothek erstellt, in der die Bewegungsdateien abgelegt sind. Zur Steuerung des Charakters kann die von Charamel entwickelte Skriptsprache CharaScript eingesetzt werden. So liegen auch CharaScript Dateien mit der Endung CS vor (in kompilierter Form CSC).

Die CharActor Technologie setzt sich aus mehreren Modulen zusammen. Das CharActor Basis Modul stellt grundlegende Funktionalitäten, wie z.B. das Rendering der 3D-Szene, zur Verfügung. Durch die Render-Engine werden die in Echtzeit dargestellten Animationen errechnet. Die Behaviour-Engine steuert das Verhalten des Charakters, Mimik und Gestik. Zur Generierung komplexer Bewegungsabläufe auf Basis der einzelnen Bewegungsdateien dient der Bewegungssynthesizer. Hierbei handelt es sich um eine in CharaScript entwickelte Programmlogik, die Bewegungsdateien nacheinander abspielt, passend zum momentanen Verhalten des Charakters. Neben den Bewegungsdateien können auch Sounddateien auf dem Server hinterlegt werden, die per Streamingverfahren zum Benutzer übertragen werden. Durch die integrierte Phonemanalyse bzw. Audioanalyse ist das Plugin in der Lage, Lippensynchronisation zum Ton zu realisieren. Hierbei werden vordefinierte Mundstellungen (Morphtargets) passend zum Ton angesteuert. Wann welches Morphtarget anzusteuern ist, wird bei Audiodateien durch Audioanalyse, bei TTS-Audio durch Phonemanalsyse ermittelt.[51]

Das CharActor TTS Modul bietet die Möglichkeit, den Charakter geschriebene Texte sprechen zu lassen. Das TTS Modul ist kein eigenständiges TTS-System, sondern eine Schnittstelle zwischen CharActor und einem TTS-System eines beliebigen anderen Anbieters. Es ist möglich, clientseitige wie auch serverseitige TTS-Systeme einzusetzen. Bei clientseitigen TTS-Systemen kann das Plugin die Installation auf dem Benutzerrechner übernehmen.

Mit dem CharActor Live Modul ist es möglich, den Charakter über einen Schauspieler zu steuern. Die Bewegungs- und Sprachdaten werden dabei live über das Internet auf den Charakter übertragen. Ein Anwendungsfeld dieses Moduls könnte die Beratung durch eine reale Person sein, falls die Wissensbasis des Avatars erschöpft ist. Auch die komplexeste Wissensbasis wird bei sehr vielen und detaillierten Fragen einmal am Ende sein und keine Antworten mehr parat haben. An diesem Punkt kann sich dann ein Mensch in die „Hülle“ des Avatars begeben und das Gespräch weiter-

führen. Der Benutzer hat den Eindruck eines fließenden Übergangs des Gesprächs, bzw. er merkt optimalerweise gar nicht, dass er sich nicht mehr mit einem Computer, sondern mit einem Menschen in Gestalt eines Avatars unterhält.

CharaScript ist die interne Skriptsprache des CharActor Plug-ins zur Steuerung von dreidimensionalen Charakteren. Eine passende Assoziation zur Erklärung der Aufgabe und Funktionalität ist das Drehbuch eines Films. Beim Film schreibt das Drehbuch dem Schauspieler vor, wann er was zu tun und zu sagen hat. Der Charakter im CharActor Plug-in wird durch CharaScript auf ähnliche Art und Weise gesteuert. Mit CharaScript hat der Programmierer die volle Kontrolle über alle Funktionen des CharActor Plug-ins und ist so in der Lage, hochwertige Charaktere im Internet zu schaffen. CharScript verwendet ein objektorientiertes Design. Auf die genauen Merkmale der Sprache und der Skripte, wie sie in einem Avatar-System Verwendung finden, wird in den folgenden Kapiteln noch näher eingegangen.

Die Firma Charamel hat selbst schon zahlreiche 3D Charaktere im Internet umgesetzt. Ein Beispiel ist der freche Fisch „Bubbles“, der in der Lage ist, auf Interaktion des Benutzers zu reagieren. So erzählt er beim Klick auf bestimmte Gegenstände Witze, oder aber liest das Horoskop vor. Da die CharActor Technologie ganz speziell für den Einsatz mit dreidimensionalen Charakteren im Internet konzipiert wurde,

Abbildung 15: CharActor Charakter: Bubbles[T]

[T] Bubbles ist online im Internet zu sehen: “URL: http://www.bubbleslife.com [Stand: 02.09.2002]”.

kann sie Vorteile aufweisen, die einen Einsatz der Software in einem Avatar-System unbedingt empfehlenswert machen. Lippensynchrone Mundbewegungen durch Phonemanalyse, die Möglichkeit den Charakter per CharaScript zu steuern, die bereits vollständig implementierte Schnittstelle zu client- wie serverseitigen TTS-Systemen – diese Faktoren bescheinigen CharActor einen klaren technologischen Vorsprung und so macht es aufgrund der vielen Vorteile auch trotz der bislang niedrigen Verbreitungszahlen Sinn, mit CharActor zu planen.

[1] Vgl. Kiwilogic: LinguBot Creator und Web Engine – Technische Daten und Systemvoraussetzungen.
Online in Internet: „URL: http://www.kiwilogic.de/linebreak/mod/netmedia_pdf/data/Lingubot%20Technologie%203.Technische%20Daten.pdf [Stand: 01.08.2002]“.
[2] Vgl. Kiwilogic: KiwiLogic LinguBot Schnittstellen und ihre Einsatzmöglichkeiten.
Online in Internet: „URL: http://www.kiwilogic.de/linebreak/mod/netmedia_pdf/data/Lingubot%20Technologie%204.%20Schnittstellen2.pdf [Stand: 01.08.2002]“.
[3] Quelle: Kiwilogic: KiwiLogic LinguBot Schnittstellen und ihre Einsatzmöglichkeiten.
Online in Internet: „URL: http://www.kiwilogic.de/linebreak/mod/netmedia_pdf/data/Lingubot%20Technologie%204.%20Schnittstellen2.pdf [Stand: 01.08.2002]“.
[4] Vgl. Gülsdorff, Björn: Gespräch vom 03.06.2002.
[5] Quelle: Vary, Peter / Heute, Ulrich / Hess, Wolfgang: Digitale Sprachsignalverarbeitung.
Stuttgart 1998, S. 6.
[6] Vgl. Vary, Peter / Heute, Ulrich / Hess, Wolfgang: Digitale Sprachsignalverarbeitung.
Stuttgart 1998, S. 6.
[7] Vgl. ebenda, S. 40.
[8] Vgl. ebenda, S.41.
[9] Quelle: Vary, Peter / Heute, Ulrich / Hess, Wolfgang: Digitale Sprachsignalverarbeitung.
Stuttgart 1998, S. 6.
[10] Vgl. Bennett , Graeme: MacOS 8.5 - Is Apple thinking differently?
Online in Internet: “URL: http://thetechnozone.com/macbuyersguide/software/system/MacOS85.html [Stand: 14.08.2002]”.
[11] Vgl. Apple Computer, Inc.: PlainTalk 1.5.4 Document and Software.
Online in Internet: “URL: http://docs.info.apple.com/article.html?artnum=60338&SaveKCWindowURL=http%3A%2F%2Fkbase.info.apple.com%2Fcgi-bin%2FWebObjects%2Fkbase.woa%2Fwa%2FSaveKCToHomePage&searchMode=Expert&kbhost=kbase.info.apple.com&showButton=false&randomValue=100&showSurvey=true&sessionID=anonymous|14046671 [Stand: 14.08.2002]”.
[12] Quinn, G. / Wang, H.-P. / Martinez, D. / Bourne, P.E. : Developing Protein Documentaries And Other Multimedia Presentations For Molecular Biology - Update on client-side speech synthesis.
Online in Internet: “URL: http://www.sdsc.edu/pb/papers/psb99_footnote.pdf [Stand: 14.08.2002]”.
[13] Vgl. MBROLA Project Development Team: The MBROLA Project - Towards a Freely Available Multilingual Speech Synthesizer.
Online in Internet: “URL: http://tcts.fpms.ac.be/synthesis/mbrola/mbrola_entrypage.html [Stand: 15.08.2002]”.
[14] Vgl. Institut für Kommunikationsforschung und Phonetik der Universität Bonn: HADIFIX.
Online in Internet: „URL: http://www.ikp.uni-bonn.de/~tpo/Hadifix.html [Stand: 15.08.2002]“.
[15] Vgl. ebenda.
[16] Vgl. Elan Speech: Speech Engine.
Online in Internet: “URL: http://www.elantts.com/products/fp_speechengine.html [Stand: 26.08.2002]”.
[17] Vgl. Black, Uyless: Internet-Technologien der Zukunft.
München 1999, S. 297.
[18] Apple Computer, Inc.: QuickTime for the Web - A Hands-On Guide for Webmasters, Site Designers, and HTMLAuthors.
San Francisco 2000, S. 1.
[19] Vgl. Waller, Richard: How Big is the Internet.

Online in Internet: “URL: http://www.waller.co.uk/web.htm [Stand: 16.08.2002]”.
[20] Vgl. Apple Computer, Inc.: Apple's QuickTime 4 Surpasses 100 Million Mark.
Online in Internet: "URL: http://www.apple.com/pr/library/2000/oct/10qtmomentum.html [Stand: 16.08.2002]“.
[21] Vgl. Apple Computer, Inc.: How popular is QuickTime?
Online in Internet: “URL: http://www.apple.com/quicktime/products/qt/faq.html [Stand: 16.08.2002]”.
[22] Quelle: Apple Computer, Inc.: QuickTime for the Web - A Hands-On Guide for Webmasters, Site Designers, and HTMLAuthors.
San Francisco 2000, S. 2.
[23] Vgl. Towner, George: Discovering QuickTime – An Introduction for Windows and Macintosh Programmers.
San Francisco 1999, S. 5, 249, 257.
[24] Glaser, Rob: Präsentation und Diskussionsrunde zur Markteinführung von Helix am 22. Juli 2002 in San Francisco.
Aufzeichnung: “URL: http://play.rbn.com/?url=tornado/tornado/demand/index_other.smil &proto=rtsp [Stand: 18.08.2002]”.
[25] Heise Online: RealPlayer für UMTS-Handys.
Online in Internet: „URL: http://www.heise.de/newsticker/data/jk-28.06.00-006/ [Stand:18.08.2002]“.
[26] Quelle: RealNetworks: The Helix™ Platform.
Online in Internet: “URL: http://www.helixcommunity.org/content/platform.html [Stand: 18.08.2002]”.
[27] Vgl. Heise Online: RealNetworks macht auf Open Source und ärgert Microsoft.
Online in Internet: „URL: http://www.heise.de/newsticker/data/vza-23.07.02-000/ [Stand: 18.08.2002]“.
[28] Quelle: RealNetworks: The Helix™ Vision.
Online in Internet: “URL: http://www.helixcommunity.org/content/vision.html [Stand: 18.08.2002]”.
[29] Vgl. RealNetworks: RealProducer Plus® User's Guide.
Online in Internet: “URL: http://www.service.real.com/help/library/guides/producerplus85/htmfiles/command.htm [Stand: 18.08.2002]”.
[30] Vgl. Alvear, José: Webdeveloper.com - Guide to Streaming Multimedia.
New York 1998, S. 159.
[31] Quelle: Microsoft: Technologies & Tools - Encoder.
Online in Internet: “URL: http://www.microsoft.com/windows/windowsmedia/WM7/encoder.asp [Stand: 24.08.2002]”.
[32] Vgl. Microsoft: Windows Media Player: Advanced Scripting for Cross-Browser Functionality.
Online in Internet: “URL: http://msdn.microsoft.com/library/en-us/dnwmt/html/cross-browser.asp?frame=true#Understanding script commands [Stand: 24.08.2002]”.
[33] Quelle: WebHits: Web-Statistiken.
Online in Internet: “URL: http://www.webhits.de/deutsch/webstats.html [Stand: 24.08.2002]”.
[34] Vgl. Alvear, José: Webdeveloper.com - Guide to Streaming Multimedia.
New York 1998, S. 180.
[35] Vgl. Niessner, Andreas: VRML-Praxis: VRML 2.0 - Virtuelle Welten dreidimensional modellieren.
München 1997, S. 9.
[36] Ebenda.
[37] Vgl. Heise Online: SIGGRAPH: 3D-Standard fürs Web nimmt Formen an.
Online in Internet: „URL: http://www.heise.de/newsticker/result.xhtml?url=/newsticker/data/js-24.07.02-003/default.shtml&words=VRML [Stand: 25.08.2002]“.
[38] Quelle: Werner, Dave: Second Nature World.
Online in Internet: “URL: http://www.dave2n.com/2nw/ [Stand: 25.08.2002]”.
[39] Vgl. Hase, Hans-Lothar: Dynamische Virtuelle Welten mit VRML 2.0 - Einführung, Programme und Referenz.
Heidelberg 1997, S. 13, 14.
[40] Ebenda, S. 2.
[41] Vgl. Viewpoint: Viewpoint Experience Technology: Technical Overview.
Online in Internet: „URL: http://www.viewpoint.com/developerzone/docs/VETTechOver.pdf [Stand: 15.06.2002]“.
[42] Vgl. Viewpoint: Creating 3D Rich Media Web Applications.
Online in Internet: “URL: http://www.viewpoint.com/developerzone/docs/Create3DApps.pdf [Stand: 15.06.2002]”.
[43] Vgl. Koglin, Ilona: Visuell greifbar.
In: Page, 03/2002, S. 76.
[44] Vgl. Phillips, Lee Anne: XML - Modernes Daten- und Dokumentenmanagement.
München 2002, S. 39.
[45] Vgl. ebenda, S. 37.
[46] Quelle: Viewpoint: Viewpoint Experience Technology: XML Authoring Overview.
Online in Internet: “URL: http://www.viewpoint.com/developerzone/docs/xmloverview.pdf [Stand: 15.06.2002]”.

[47] Loviscach, Jörn: Bauchredner - Die Animationssoftware TrueSpace 5.2 bringt 3D-Köpfe automatisch zum Sprechen.
In: c't 9, 22.04.2002, S.65.
[48] Puscher, Frank: Bau dir deinen Avatar.
In: Internet World, Februar 2002, S. 87.
[49] Vgl. Charamel: Einbindung virtueller Charaktere in Webseiten mit CharActor.
Vertrauliches PDF-Dokument der Charamel GmbH, S. 3.
[50] Vgl. ebenda.
[51] Vgl. ebenda, S. 7.

5. Systemarchitektur

In den vorigen Kapiteln wurden verschiedene Technologien vorgestellt, die potentiell in einem Avatar-System eingesetzt werden können. Es wurde analysiert, wo Vor- und Nachteile liegen, wo möglicherweise notwendige Schnittstellen geschaffen werden, bzw. bereits vorhandene Schnittstellen verwendet werden können.

Es werden nun theoretische Überlegungen angestellt, wie diese einzelnen Technologien zu einem System zu kombinieren sind, wie die Architektur eines solchen Systems aussehen kann. Speziell sollen zwei Architekturen unter Verwendung unterschiedlicher Systemkomponenten entwickelt werden. Jede Architektur weist andersartige Vor- und Nachteile auf.

5.1. Architektur 1: Viewpoint, WinMedia, LinguBot

Der Aufbau dieses Systems sieht die Verwendung der Komponenten KiwiLogic LinguBot, HADIFIX, MBROLA, Windows Media, sowie der Viewpoint Technologie vor. Zudem kommen PHP und MSDOS auf Serverseite zum Einsatz, um die zentrale Steuerung zu übernehmen. Grundgedanke dieser Entwicklung ist, auf Serverseite eine Audiodatei mit integrierten Skriptkommandos dynamisch zu erzeugen. Die Datei wird daraufhin über das Internet übertragen und beim Client im Windows Media Player abgespielt. Über eine Schnittstelle kann der 3D-Charakter per Skriptkommandos gesteuert werden.

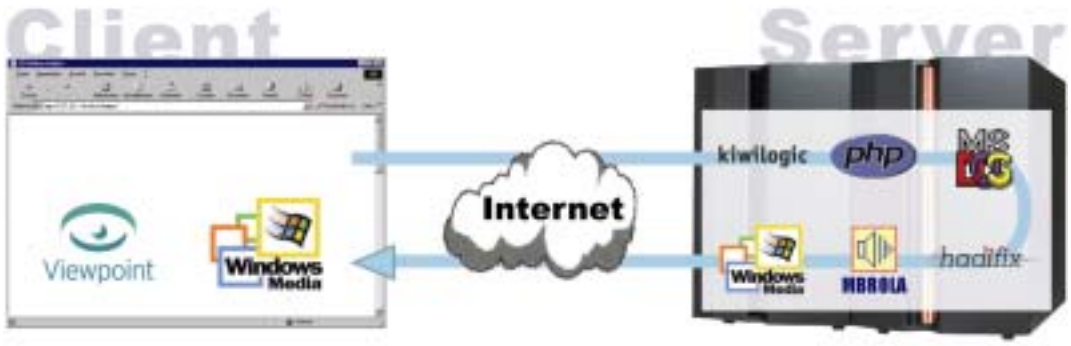

Abbildung 16: Systemarchitektur 1

Wichtiges Element bei der Erstellung einer solchen Architektur ist die genaue Planung der Kommunikation zwischen den einzelnen Komponenten. Schon jetzt muss geklärt werden, welche Schnittstellen genutzt werden können und wo eigene zu entwickeln sind. Im Folgenden wird daher auf Client-, wie auch auf Serverseite das gesamte Zusammenspiel der Software theoretisch durchdacht. Es gilt sicherzustellen, dass die Systemarchitektur prinzipiell funktioniert und bei der Implementierung keine unüberwindbaren Hindernisse auftauchen.

5.1.1. Architektur auf Clientseite

Der Ablauf auf Clientseite wird wie folgt aussehen. Der Benutzer trägt seine Frage in ein HTML-Formular ein und sendet dieses ab. Eigentlich würde es nun ausreichen, dem Client die fertige ASF-Datei mit integrierten Skriptkommandos zurückzuliefern, doch LinguBot bietet leider nur die Schnittstelle über ein HTML-Template. Das bedeutet, dass die Antwort, welche LinguBot ermittelt hat, zunächst im HTML-Code in Textform an den Client zurückgegeben wird. Dort muss daraufhin ein automatischer Prozess in JavaScript gestartet werden, welcher die Antwort wieder an den Server zur TTS-Konvertierung schickt. Ist der Text in Audio konvertiert, so kann die Audiodatei in den Windows Media Player geladen und dort abgespielt werden. Das Laden der neuen Audiodatei kann per JavaScript erfolgen, da Microsoft mit dem Media Player Plug-in ein JavaScript Interface für den Zugriff auf Funktionalitäten des Players zur Verfügung stellt.

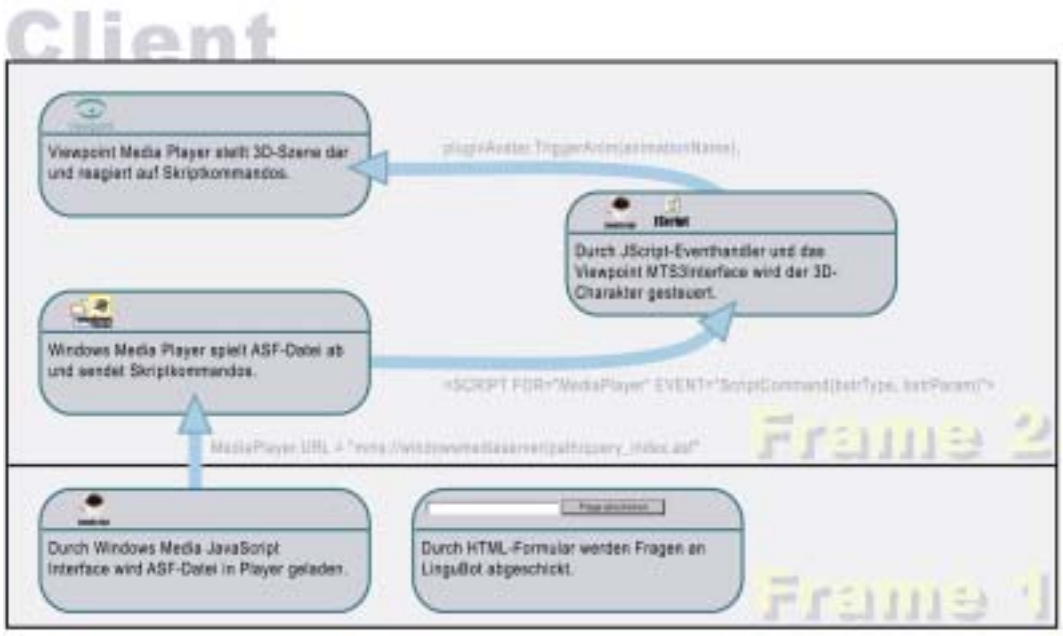

Abbildung 17: Systemarchitektur 1 - Clientseite

Offensichtlich ist es notwendig, das Browserfenster des Clients in zwei Frames aufzuteilen. Die Kommunikation, der Datenaustausch zwischen Webserver und Browser funktioniert beim Übertragungsprotokoll HTTP immer nach dem Pull-Prinzip. Es ist dem Server nicht möglich, aktiv Daten an den Client zu senden. Vielmehr muss der Server immer auf Anfragen des Clients warten, um diese zu beantworten. Zum Datenaustausch muss der Client in der Regel ein Dokument oder Skript des Servers laden. Es gibt also einen Frame, der zur Kommunikation zwischen Browser und Server dient. In diesem Frame (siehe Abbildung 17: Frame 1) befindet sich das HTML-Formular, sowie einige JavaScript-Funktionalitäten. Per JavaScript wird aus dem Frame die Antwort an den Webserver geschickt, um in Audio konvertiert zu werden. Außerdem befindet sich hier die Funktion, welche die Audiodatei schließlich in den Windows Media Player lädt. In einem zweiten Frame (siehe Abbildung 17: Frame 2) befinden sich die Elemente, die nicht ständig neu geladen werden sollen. Das ist zum einen der Windows Media Player, zum anderen der Viewpoint Media Player.

Der Viewpoint Media Player ist zur 3D-Darstellung des Charakters vorgesehen. In der MTX-Datei, welche den Viewpoint Charakter letztlich steuert, sind verschiedene Animationen anzulegen. So kann es beispielsweise A-

nimationen während des Sprechens geben, welche abgespielt werden, wenn nichts passiert, oder aber Animationen, die ständig wiederholt werden, wie z. B. die Mundbewegungen während des Redens. Wichtig dabei ist, dass alle diese Animationen in ein und derselben Anfangs- bzw. Endposition beginnen und auch enden. Nur so ist ein fließender Übergang zwischen verschiedenen Animationen zu realisieren. Es ist möglich, die Animationen, die in der MTX-Datei definiert sind, per JavaScript zu starten. Hierfür stellt Viewpoint das MTS3Interface zur Verfügung. Es handelt sich dabei um eine JavaScript-Schnittstelle für den Viewpoint Media Player. Die Funktion `TriggerAnim('reden')` kann eine Animation starten, in diesem Fall die Animation mit dem Namen „reden“, welche die Bewegungen des Mundes beinhaltet. Auf diese Weise können Animationen nacheinander abgespielt werden. Sollte es jedoch notwendig sein, Animationen zu entwickeln, die bewusst in einer nicht identischen Start- und Endposition beginnen und enden, so ist es notwendig, die aktuellen Positionen über Statusvariablen in JavaScript zu speichern. Dreht der Avatar seinen Kopf nach links, muss in einer Variable gespeichert werden, dass sich der Kopf nun links befindet. Die nächste Kopfanimation kann dann keinesfalls wieder die Drehung nach links, sondern eher eine Kopfdrehung nach rechts sein. Werden Positionsänderungen nicht beachtet, kann es zu Sprüngen im Bewegungsablauf des Charakters kommen.

Weiter muss ein Eventhandler in der Sprache JScript in die Seite integriert werden. Dieser Eventhandler fängt die Skriptkommandos des Windows Media Player ab und ruft daraufhin die passenden JavaScript-Funktionen des MTS3Interface. So kann der Avatar auf Basis des Audiomaterials gesteuert und Synchronisation zwischen Ton und Mundbewegung erzielt werden.

5.1.2. Architektur auf Serverseite

Aufgabe des Servers ist es, eine passende Antwort auf die Frage des Benutzers zu ermitteln, diese möglichst schnell in Audio zu konvertieren und

schließlich an den Client zu senden. Der erste Schritt wird durch das Dialogsystem KiwiLogic LinguBot realisiert. Die Software erhält als Input die Frage des Benutzers und sucht daraufhin in der aktuell vorliegenden Wissensbasis nach einer möglichst passenden Antwort. Es ist nicht zu umgehen, dass der LinguBot die Antwort in einer HTML-Seite an den Client zurückgibt. Daher enthält diese Seite nichts als die ermittelte Antwort und einen JavaScript-Befehl, welcher die Antwort unmittelbar an ein PHP-Serverskript auf dem Server zurückschickt. Dieses Programm ist der eigentliche Kern der Serverprogrammierung. Hier werden alle Prozesse verwaltet und die Konvertierung des Textes in Ton gesteuert.

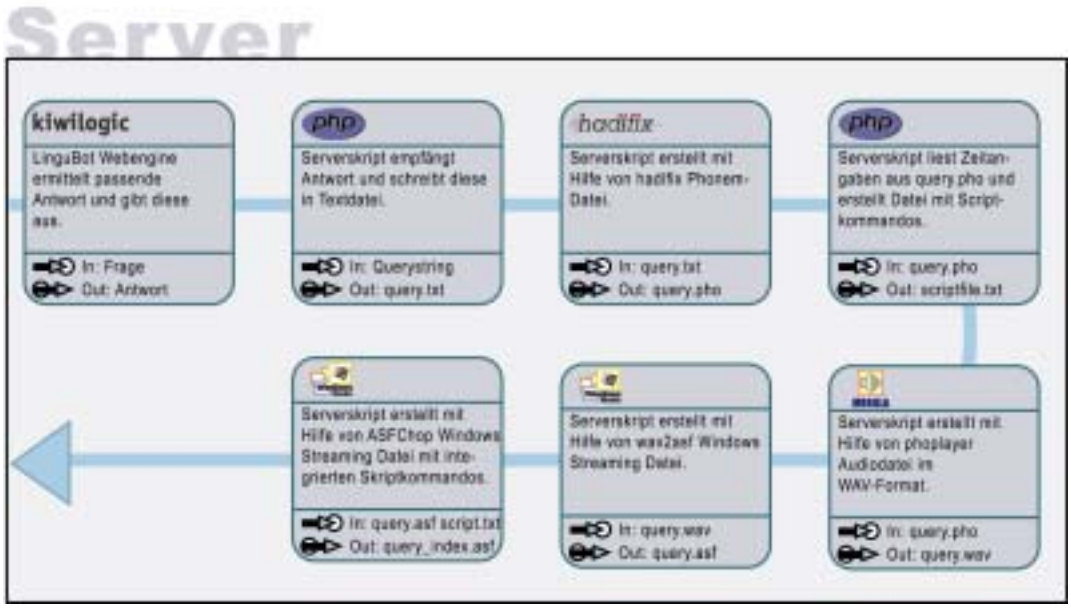

Abbildung 18: Systemarchitektur 1 - Serverseite

Zunächst empfängt das Serverskript die Antwort und schreibt sie zur späteren Weiterverarbeitung in eine Textdatei (query.txt). Mit Hilfe des Programms txt2pho von HADIFIX erstellt das Skript aus der Textdatei eine Phonemdatei (query.pho). Diese Datei enthält die einzelnen Phoneme, sowie Angaben zu Zeit, Phrasierung, etc. Da txt2pho nicht direkt aus dem Serverskript heraus aufgerufen werden kann, ist ein kleines Hilfsprogramm nötig. In diesem Falle einer Windows Plattform verwendet man eine Batch-Datei (Dateiendung BAT), das Pendant unter Linux oder Unix-Derivaten ist ein Shellskript. Der nächste Schritt könnte nun die Erstellung einer Skriptdatei (scriptfile.txt) sein, welche die Skriptkommandos zur Steuerung des

Avatars enthält. Eine sehr einfache Skriptdatei enthält nur zwei Kommandos, nämlich `startTalk` zu Beginn und `stopTalk` zum Ende. Um den richtigen Zeitpunkt für `stopTalk` zu ermitteln, muss aber zunächst die Gesamtdauer der noch zu erzeugenden Audiodatei festgestellt werden. Dies ist möglich, indem das Serverskript die zuvor erzeugte Phonemdatei ausliest und die Zeitangaben der einzelnen Phoneme aufsummiert. Mit der somit festgestellten Gesamtzeit kann dann die Skriptdatei erzeugt werden.

Die nächsten Schritte beinhalten die eigentliche Audiogenerierung, sowie die Aufbereitung der Audiodatei. Durch Aufruf des Programms phoplayer, welches Element des MBROLA Speech Synthesizers ist, wird mit dem Input der query.txt eine Audio-Datei im WAV-Format (query.wav) erzeugt. Diese Datei wird von wav2asf durch das Serverskript in eine ASF-Datei (query.asf) konvertiert. Um schließlich noch die Skriptkommandos in die ASF-Datei zu integrieren, muss das Programm ASFChop ausgeführt werden. ASFChop erhält als Input die Dateien query.asf und scriptfile.txt. Hiermit erstellt es eine neue ASF-Datei namens query_index.asf. Die Audioerzeugung ist nun abgeschlossen. Das Serverskript liefert dem Client eine HTML-Antwortseite, die wieder das Formular enthält um eine neue Frage abzuschicken. Außerdem wird die Antwort hier zusätzlich in Textform angezeigt. Ein JavaScript-Befehl, der beim Laden der Seite ausgeführt wird, lädt die auf dem Server befindliche ASF-Datei in den Windows Media Player in Frame 1.

5.1.3. Stärken und Schwächen

Nicht ganz optimal an dieser Systemarchitektur ist, dass pro Frage-Antwort-Kombination zweimal zwischen Client und Server kommuniziert werden muss. Das ist jedoch nicht zu ändern, da der LinguBot lediglich ein HTML-Template zur Integration einer Schnittstelle anbietet. Die Antwort des LinguBots wird also in jedem Fall als HTML-Output an den Client zurückgegeben, bevor andere Softwarekomponenten mit der ermittelten Antwort arbeiten können. Wirklich problematisch ist das allerdings auf-

grund der sehr geringen Datenmengen, die diese Kommunikation benötigt, nicht. Wie bereits erwähnt, enthält der Output des LinguBot im Wesentlichen die Antwort und einen JavaScript-Befehl. Es handelt sich also um Datenmengen, die auch für langsame Internetverbindungen kein Problem darstellen.

Ein weiteres Problem könnte die Synchronisation zwischen Windows Media Player und Viewpoint Plug-in über das MTS3Interface darstellen. Erfahrungsgemäß ist die Geschwindigkeit, mit der JavaScript-Befehle von Plug-ins fehlerfrei empfangen und verarbeitet werden können, nicht allzu hoch. Da die Lippensynchronität nicht durch das Ansteuern einzelner Morphtargets erzielt wird, sondern über eine wiederholbare Animation, sollte jedoch auch das in der Praxis kein großes Problem darstellen.

Vorteilhaft ist sicherlich die Verwendung des Windows Media Players im Browser des Clients. Das Plug-in ist weit verbreitet und die Art des Einsatzes in diesem System ermöglicht auch die alternative Verwendung von QuickTime oder Real/Helix. Die erforderliche Funktionalität kann von jedem der drei Plug-ins erbracht werden. Für diese Architektur wurde beispielhaft der Windows Media Player gewählt, da er momentan die größten Verbreitungszahlen aufweist. Auch bei der Viewpoint Technologie handelt es sich um ein Produkt, von dem ein rascher Anstieg der Verbreitungszahlen im Zuge der Integration in AOL 7 erwartet wird. Die Qualität des Renderings ist zudem ausgesprochen gut.

Das System ist somit theoretisch skizziert, die einzelnen Schnittstellen, die zur Kommunikation benötigt werden, sind sichergestellt. Auf theoretischer Ebene ist alles getan, was ein Gelingen der Implementierung sicherstellen soll.

5.2. Architektur 2: CharActor, LinguBot, Elan

Die zweite Architektur, die im Rahmen dieser Arbeit entwickelt wurde, sieht die Verwendung der CharActor Technologie vor. Als TTS-System, das serverseitig an CharActor angebunden ist, kommt Elan Speech zum Einsatz. CharActor bietet auf beiden Enden des Systems, also auf dem Server wie dem Client, große Vorteile, da die Technologie einige Bereiche abdeckt, für die in Architektur 1 verschiedene Komponenten notwendig waren.

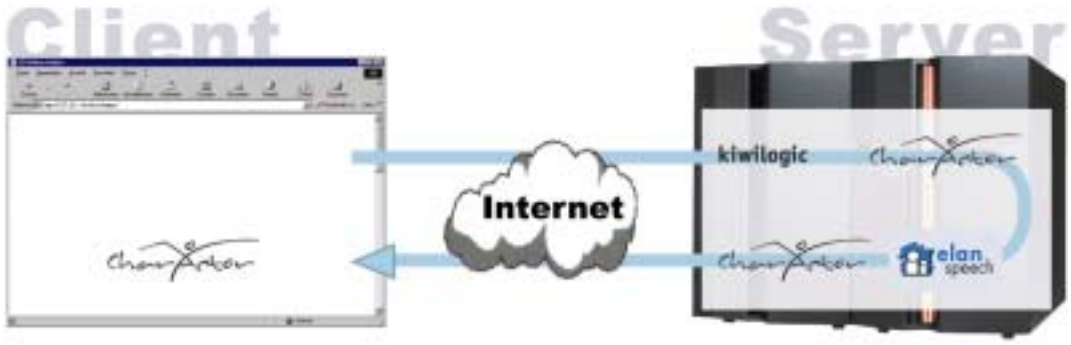

Abbildung 19: Systemarchitektur 2

Die Idee dieser Architektur unterscheidet sich von der Ersten in einigen Punkten. Eine Audio-Datei mit integrierten Skriptkommandos wird hier nicht benötigt. In Architektur 1 besteht die Hauptaufgabe der Skriptkommandos darin, die Mundbewegungen mit dem Ton zu synchronisieren. Aufgrund der Phonemanalyse, die Bestandteil von CharActor ist, erübrigt sich das. Durch die Phonemanalyse ist das CharActor Plug-in selbständig in der Lage, Mundbewegungen über Morphtargets zu synchronisieren. Weiter kann auch auf ein zusätzliches Plug-in zur Audiowiedergabe verzichtet werden. CharActor ist in der Lage, den TTS-Ton wiederzugeben, während parallel die 3D-Szene dargestellt wird.

5.2.1. Architektur auf Clientseite

Wie bereits erwähnt, kommt diese Architektur auf Clientseite mit nur einem Plug-in aus. Doch auch hier ist es notwendig, das Browserfenster des Clients in zwei Frames aufzuteilen. Zu Anfang stellt sich der Ablauf gleich

dar wie in Architektur 1. Der Benutzer gibt die Frage in das HTML-Formular ein und schickt es ab. Die von LinguBot ermittelte Antwort wird in einer HTML-Seite zurückgegeben. Nun ist die Funktionsweise jedoch anders als im ersten Fall. Eine JavaScript-Funktion schickt die Antwort nicht zurück an ein Serverskript, sondern übergibt sie direkt an das CharActor Plug-in in Frame 2. Die weiteren Aufgaben kann CharActor daraufhin selbständig bewältigen. Die Antwort wird vom Plug-in an ein CGI-Programm auf dem Webserver übermittelt, wo die TTS-Konvertierung erfolgt.

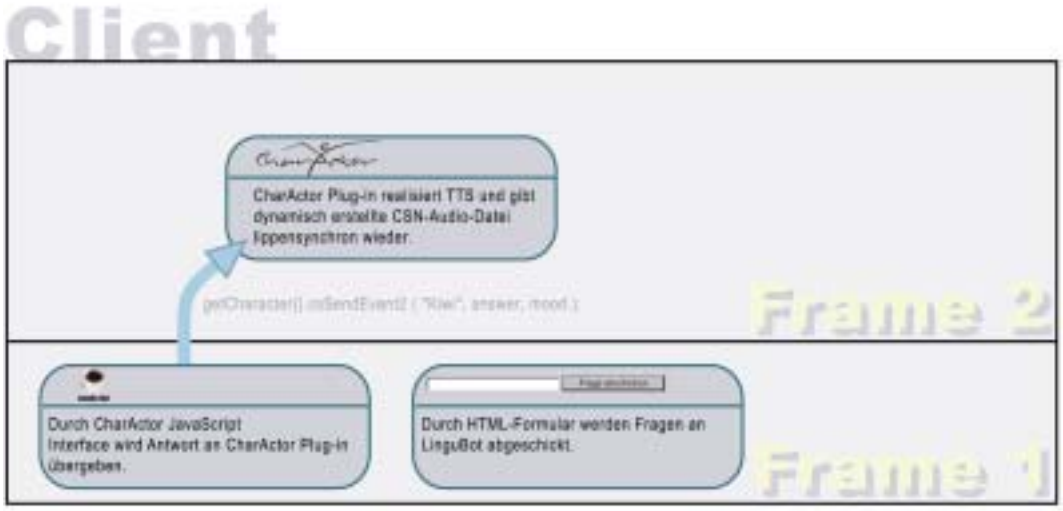

Abbildung 20: Systemarchitektur 2 – Clientseite

Der Zugriff auf das CharActor Plug-in per JavaScript ist möglich, da Charamel hierfür ein spezielles Interface geschaffen hat. Bei der Vorstellung der CharActor-Technologie wurde schon erwähnt, dass es eine interne Programmiersprache zur Steuerung des Charakters namens CharaScript gibt. Jegliche Aktion des Charakters ist innerhalb des Plug-ins mit CharaScript zu programmieren. Diese Sprache arbeitet massiv nach dem Prinzip, Events auszulösen und an anderer Stelle darauf zu reagieren. Mit JavaScript ist es möglich, ein Event auch von außerhalb des Plug-ins auszulösen, darauf kann dann mit CharaScript im Plug-in reagiert werden. In diesem Falle wird das Event `Kiwi` ausgelöst, ein in CharaScript implementierter Eventhandler names `onKiwi` reagiert darauf. Die Aufgabe die-

ses Eventhandlers ist in erster Linie den empfangenen Text, die Antwort, mit dem CharaScript-Befehl `TTS.say()` an das serverseitige TTS-System weiterzuleiten, den TTS-Prozess und die Phonemanalyse einzuleiten und den Charakter schließlich sprechen zu lassen.

5.2.2. Architektur auf Serverseite

Auch auf Serverseite kommt Architektur 2 mit weniger Komponenten als Architektur 1 aus. Grund dafür ist, dass viel Funktionalität in CharActor bereits eingearbeitet ist und somit nicht alles selbst entwickelt werden muss.

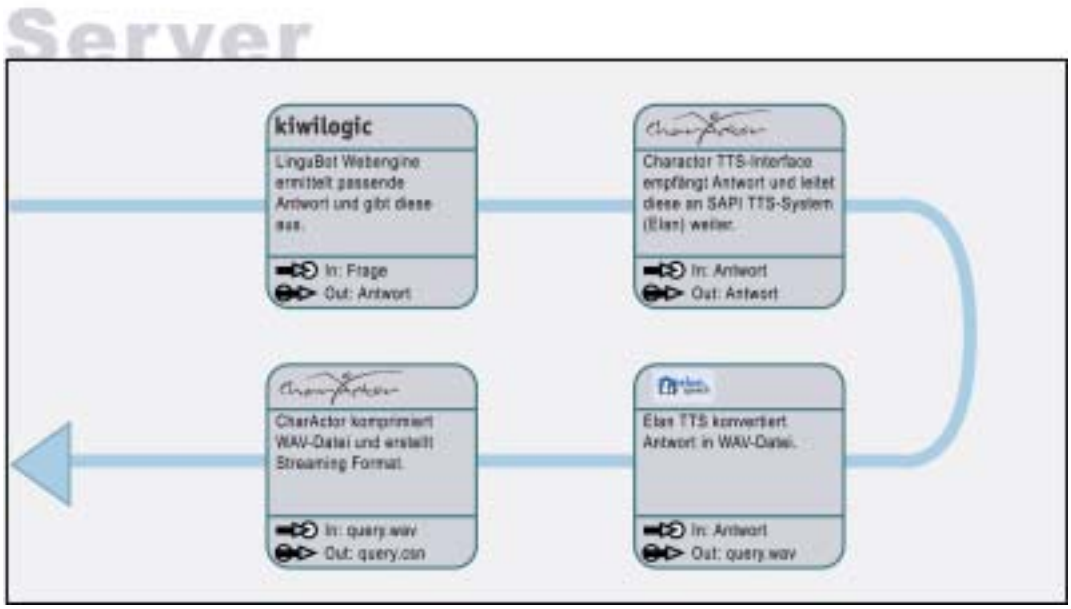

Abbildung 20: Systemarchitektur 2 – Serverseite

Das Plug-in schickt die Antwort an ein CGI-Programm (tts.cgi) auf dem Webserver. tts.cgi ruft wiederum ein Zusatzprogramm namens TTSManager.exe, um hiermit auf das installierte SAPI-TTS-System, in diesem Fall Elan Speech, zuzugreifen. TTSManager.exe ist eine Software, welche die verschiedenen TTS-Systeme für CharActor verwaltet. Das Programm gibt die Antwort des LinguBot an das richtige TTS-System weiter. Welches der auf dem Server befindlichen Systeme verwendet werden soll, wird in einer Konfigurationsdatei des jeweiligen Charakters namens behave.cfg definiert. Hier sind auch Parameter, wie Stimmhöhe (Pitch Baseline), Ge-

schwindigkeit, Lautstärke und der Name der Stimme anzugeben. Elan Speech erzeugt mit dem Input eine Audiodatei im WAV-Format und gibt diese an TTSManager zurück. Dieser erzeugt aus der WAV-Datei drei unterschiedlich stark komprimierte Sounddateien im CharActor Sound-Format. Die drei Sounddateien sind MP3-komprimiert und weisen die Datenraten 8kbit/sec, 16 kbit/sec, 32 kbit/sec auf.[1] Die drei Dateien werden auf dem Server abgelegt. Das Plug-in lädt nun je nach Einstellung eine der drei Dateien im Streaming-Modus.

5.2.3. Stärken und Schwächen

Wie bereits in Architektur 1 festgestellt, ist auch hier von Nachteil, dass eine zweimalige Kommunikation zwischen Client und Server pro Frage-Antwort-Kombination notwendig ist. Aufgrund des niedrigen Datenaufkommens kann man jedoch nicht von einem großen Nachteil sprechen. Die Verwendung von CharActor bringt diesbezüglich noch den Vorteil mit sich, dass der zweite Kommunikationsschritt nicht das Versenden einer kompletten HTML-Datei (mit File-Header, etc.) mit sich bringt. Das CharActor-Plugin ist selbständig in der Lage, Daten mit tts.cgi auszutauschen, dabei kommt nur ein minimales Datenaufkommen zustande.

Synchronisation zwischen Ton und Mundbewegungen ist bei CharActor durch Phonemanalyse und Morphtargets sichergestellt. Die generierten Mundbewegungen wirken wesentlich natürlicher als das eine Animationsschleife leisten kann. Die ständige Steuerung des Plug-ins von außen durch JavaScript entfällt, hierfür steht die interne Skriptsprache CharaScript zur Verfügung.

Ganz klarer Vorteil dieser Struktur ist die schlankere Architektur auf Client- wie Serverseite. CharActor bietet ein breites Spektrum an Funktionalität, speziell zur Verwendung mit einem Avatar und so kann auf einige Komponenten, die für Architektur 1 zwingend erforderlich waren, hier verzichtet werden.

Auch dieses System ist somit theoretisch durchgeplant und alle Schnittstellen scheinen so zur Verfügung zu stehen, wie sie benötigt werden. Einer erfolgreichen Implementierung sollte daher nichts mehr im Wege stehen.

[1] Vgl. Reinecke, Alexander: E-Mail vom 26.08.2002 10:41, Betreff: AW: elan tts.

6. Implementierung des Systems

Wer Software implementiert, weiß, dass zwischen theoretischer Planung und praktischer Umsetzung ein großer Unterschied besteht und dass erst in der Umsetzung Probleme auftauchen können, die so nicht vorhersehbar sind. Eine möglichst perfekte Planung eines Projektes, wie das der Entwicklung eines Avatar-Systems, ist daher umso mehr unverzichtbare Grundlage für die Produktion. Im vorigen Kapitel wurden alle theoretischen Eventualitäten durchgespielt, das Zusammenspiel der Schnittstellen durchdacht und mögliche Schwachstellen aufgezeigt. Für beide Architekturen lautet das Ergebnis, dass einer Implementierung prinzipiell nichts im Wege steht. Ob dies auch in der Praxis zutrifft, soll im Folgenden analysiert werden. Beide Architekturen werden daher praktisch umgesetzt.

6.1. Architektur 1

Es gilt ein System entsprechend der ersten Architektur in der Praxis zu entwickeln. Die Implementierung ist in zwei Bereiche gegliedert. Zum einen muss die technische Grundlage für das System geschaffen werden. Ein Server mit der benötigten Software ist einzurichten und muss konfiguriert werden. Der zweite Schritt ist die Entwicklung des eigentlichen Systems, also die Erstellung der Software. Insbesondere wird auf relevante Einzelheiten der praktischen Umsetzung, sowie auf Problematiken, die sich erst während dieser herauskristallisierten, eingegangen werden.

6.1.1. Implementierung

Die erste Frage bei Beginn der Implementierung ist die nach der Wahl des richtigen Server-Betriebssystems. In diesem Falle bleiben nicht viele Möglichkeiten, da mit Windows Media Dateien gearbeitet werden soll. Programme wie wav2asf oder ASFChop sind nur für Windows-Plattformen erhältlich und so ist man auf die Verwendung eines Windows-Servers angewiesen. Das Testsystem, auf welchem dieser Prototyp entwickelt wird, ist daher ein Server unter Windows 98. Als Hauptspeicher stehen 192 MB zur Verfügung und der Rechner ist mit einem Pentium II Prozessor ausgestattet, der mit 333 MHz getaktet wird.

Um den Rechner als Webserver einzusetzen, ist die entsprechende Software zu installieren. Apache 1.3.23 (Win32) und PHP Version 4.1.2 statten den Rechner mit der nötigen Basisfunktionalität aus. Die beiden Softwarekomponenten sind unkompliziert zu installieren, für beide liegen binäre Installationsprogramme vor. Nach der Installation muss Apache durch die Datei httpd.conf konfiguriert werden. Hier sind lediglich einige Parameter für die Verwendung von PHP einzutragen bzw. anzupassen. Durch die beiden Zeilen

```
AddType application/x-httpd-php .php
Action application/x-httpd-php "/php/php.exe"
```

wird die Anwendung php.exe zum Parsen von PHP-Dateien (Dateiendung PHP) in Apache registriert. Zur Verwendung der LinguBot Web Engine, unterhalb des Root-Verzeichnisses des Webservers ist zudem die Angabe `AddHandler cgi-script .exe .cgi` hinzuzufügen. Diese bewirkt, dass auch EXE-Dateien als CGI-Programme ausgeführt werden können. PHP ist mittels der Datei php.ini zu konfigurieren, welche sich unter Windows im Verzeichnis C:\Windows\ befindet. Spezielle Angaben sind hier nicht notwendig.

Nächster Schritt ist die Installation der einzelnen Komponenten, die auf dem Webserver benötigt werden. Bei dieser Architektur sind das HADIFIX, der MBROLA Speech Synthesizer und das Windows Media Ressource Kit, welches die Programme wav2asf und ASFChop enthält. Auch diese Instal-

lationsprozesse gestalten sich unkompliziert, da fertige Installationsroutinen vorliegen. HADIFIX und Windows Media sind direkt nach der Installation betriebsbereit, für den MBROLA Speech Synthesizer muss noch eine der zahlreichen Stimmen im Control Panel registriert werden. Die Stimme „de2“ erscheint am hochwertigsten und somit für diese Anwendung am besten geeignet.

Abbildung 22: Avatar: Wiseman – Systemarchitektur 1

Nachdem das System fertig installiert, konfiguriert und somit betriebsbereit ist, folgt der wesentlich aufwändigere Teil, die Erstellung des eigentlichen Avatar-Systems, die Programmierung der Software. Wie bereits erwähnt, gibt es ein zentrales PHP-Skript, welches alle Vorgänge steuert. Das Fragenformular ist gleichzeitig der HTML-Output des Skriptes. Sendet der Benutzer das Formular ab, so startet er damit auf Serverseite einen Prozess, welcher die Schritte Antwortermittlung, TTS, Audiokonvertierung, Integration von Skriptkommandos und Auslieferung der Daten an den Client beinhaltet. Ziel des Formulars ist die LinguBot Web Engine (`action="cgi-bin/localbot.exe"`). Dies ist das Programm, das die Frage des Benutzers zur Ermittlung einer passenden Antwort erhält. Output der Web Engine ist eine einfache HTML-Seite, welche die ermittelte Antwort und ein JavaScript-Kommando enthält. Dieses Kommando schickt die Antwort direkt an das PHP-Skript auf dem Server zur weiteren Verarbeitung.

Zunächst schreibt das PHP-Skript die von der Webengine ermittelte Antwort in eine Textdatei namens query.txt

```
$fp = fopen ("e:\\webserver\\audio\\query.txt", "w");
fputs($fp, $query);
fclose($fp);
```

Mit der Textdatei wird daraufhin eine Phonemdatei erzeugt. Hierfür ist das Programm txt2pho notwendig. Dieses Programm erhält als Input die Textdatei, es erzeugt damit die Phonemdatei query.pho.

```
txt2pho e:\webserver\audio\query.txt
e:\webserver\audio\query.pho
```

Nächster Schritt ist die Ermittlung der Gesamtdauer der auszugebenden Antwort mit Hilfe von query.pho. Diese Datei enthält neben diversen Angaben zur Steuerung des MBROLA Speech Systhesizers die Dauer eines jeden Phonems in Millisekunden. Die Datei liegt nicht binär, sondern im ASCII-Format vor und kann somit von PHP ausgelesen werden. Jedes Phonem belegt in query.pho eine eigene Zeile.

```
y: 36  42  85  97  85
```

Die erste Angabe ist dabei die Kennung des Phonems, die zweite dessen Dauer. Durch folgende PHP-Zeilen kann die Gesamtdauer einer Antwort ermittelt werden.

```
$time = 0;
$fp = fopen ("e:\\webserver\\audio\\query.pho", "r");
while (!feof ($fp)) {
    $buffer = fgets($fp, 4096);
    if(strlen($buffer)>0) $thisTime = substr($buffer,
    strpos($buffer, " ")+1,strpos($buffer, " ",
    strpos($buffer, " ")+1)-2);
    $time += $thisTime;
}
fclose ($fp);
```

Die Ermittlung der Dauer ist notwendig, um eine Skriptdatei zu erzeugen, welche Skriptkommandos zur Steuerung des 3D-Plug-ins enthält. Es ist weiter möglich, die Phonemdatei nach unterschiedlichsten Kriterien zu analysieren. So kann man z.B. nach Phonemkombinationen suchen und

diese mit Animationen koppeln. Die Animation „Kopfschütteln“ könnte so mit den Phonemen verknüpft werden, die das Wort „Nein“ repräsentieren.

Die einfachste Form einer Skriptdatei soll hier veranschaulicht werden. Diese Datei enthält lediglich einen Start- und einen Stoppbefehl. Diese steuern die Animationsschleife der Mundbewegung.

```
$scriptfile=fopen("$PathToFolder\\scriptfile.txt","w");
fputs($scriptfile,"start_script_table\n");
fputs($scriptfile,"00:00:00.0 startTalk XparaX\n");
//time minus because of silence in the end (483msec)
$time -= 600;
$scriptTime_stop=timeInPhoFormat($time);
$scriptInsert="$scriptTime_stop stopTalk XparaX\n";
fputs($scriptfile,$scriptInsert);
fputs($scriptfile,"end_script_table");
```

Die Praxis zeigt nun, dass txt2pho an das Ende einer jeden Phonemdatei eine Pause einfügt. Die zu erzeugende Audiodatei enthält somit noch eine kurze Stille am Ende. Daher wird die Zeitangabe für den StopTalk Befehl um 600 Millisekunden verringert. Die Funktion `timeInPhoFormat()` wandelt die Zeitangabe in das passende Format zur Verwendung in der Skriptdatei.

Da an diesem Punkt Skriptdatei und Phonemdatei fertig vorliegen, bildet der nächste Schritt die Audioerzeugung und Konvertierung. Das Programm phoplayer des MBROLA Speech Synthesizers erzeugt auf Basis der Phonemdatei query.pho eine Audiodatei im Wav-Format (query.wav).

```
exec("phoplayer database=D:\database\de2\de2
/O=e:\webserver\audio\query.wav
/T=WAV e:\webserver\audio\query.pho");
```

Diese wird durch das Programm wav2asf, welches Element des Windows Media Ressouce Kit ist, in das Windows Streaming Format gewandelt.

```
exec("wavtoasf -in e:\webserver\audio\query.wav
-out e:\webserver\audio\\".$filename.".asf");
```

Letzter Schritt ist schließlich die Integration der Skriptkommandos mittels ASFChop.

```
exec("ASFChop -in e:\webserver\audio\\".$filename.".asf
-out e:\webserver\audio\\".$filename."_index.asf
-script scriptfile.txt");
```

Eine Sache, die sich erst bei einem Praxistest herausgestellt hat, ist die Notwendigkeit eindeutiger Dateinamen für die ASF-Dateien. Werden diese Dateien per HTTP an den Client ausgeliefert und nicht über ein wirkliches Streaming Protokoll, so passiert ein Caching der Dateien auf Clientseite. Der Client merkt somit nicht mehr, dass sich die Audiodateien von Antwort zu Antwort ändern, er gibt immer dieselbe Antwort aus. Durch eine Kombination von Datum und Zeitangabe wird daher ein eindeutiger Dateinamen erzwungen, Caching findet nicht mehr statt.

Die Audioerzeugung ist somit abgeschlossen, das PHP-Skript liefert das HTML-Formular als Output zurück an den Client. In diesem Formular befindet sich ein JavaScript-Befehl, durch den die soeben erzeugte ASF-Datei in das Media Player Plug-in in Frame 2 geladen wird.

```
echo "top.main.MediaPlayer.URL =
\"http://127.0.0.1/audio/".$filename."_index.asf\";\n";
```

Während das Media Player Plug-in die Audiodatei abspielt, werden die durch die Skriptdatei definierten Events abgefeuert. Ein in JScript geschriebener Eventhandler, der sich ebenfalls in Frame 2 befindet, fängt diese ab und reagiert darauf.

```
<SCRIPT FOR="MediaPlayer"
EVENT="ScriptCommand(bstrType, bstrParam)"
LANGUAGE="Jscript">
if (bstrType.toLowerCase() == "starttalk") {
    pluginAvatar.StartAnim('reden');
}
if (bstrType.toLowerCase() == "stoptalk") {
    pluginAvatar.StopAnim('reden');
    pluginAvatar.StartAnim('still');
```

```
    setTimeout("freeMPlayer()", 2000);
}
</SCRIPT>
```

Von diesem Eventhandler werden Funktionen des MTS3Interface gerufen, der Schnittstelle für Zugriff auf den Viewpoint Media Player. `StartAnim()` startet eine Animation, `StopAnim()` stoppt eine solche. Eine weitere Eigenart des Windows Media Players hat sich während intensiven Praxistests gezeigt. Von Zeit zu Zeit kommt es vor, dass der Windows Media Player keinen Zugriff mehr auf die URL Eigenschaft gestattet. Das hat zur Folge, dass Fehler auftreten und kein weiteres Audiomaterial mehr in den Player geladen werden kann. Abhilfe schafft eine Funktion (`freeMPlayer()`), welche die URL des Players auf „X“ setzt, also praktisch löscht. Grund für dieses Verhalten ist wohl die Kombination von Windows Media Player und Viewpoint Media Player in einer HTML Seite. Setzt man die Plug-ins getrennt voneinander ein, so kommen Fehler dieser Art nicht vor.

6.1.2. Analyse und Bewertung

Es ist gelungen, das System nach Architektur 1 auch in der Praxis umzusetzen. Die theoretischen Erkenntnisse aus Kapitel 5 waren demnach korrekt und so kam es zu keinen unüberwindbaren Hindernissen, welche die Implementierung hätten verhindern können. Serverseitig funktioniert das System sehr gut, die Audioerzeugung und Integration von Skriptkommandos ist problemlos. Große Chancen bieten sich in der Erweiterung der Analyse der Phonemdatei. Hier sind praktisch kaum Grenzen gesetzt. Die Datei kann nach beliebigen Faktoren untersucht werden und daraufhin können spezielle Skriptkommandos in die Skriptdatei und somit die ASF-Datei integriert werden. Die Aktionsfähigkeit des Charakters kann somit gut ausgebaut, ein relativ natürliches Verhalten kann erzielt werden.

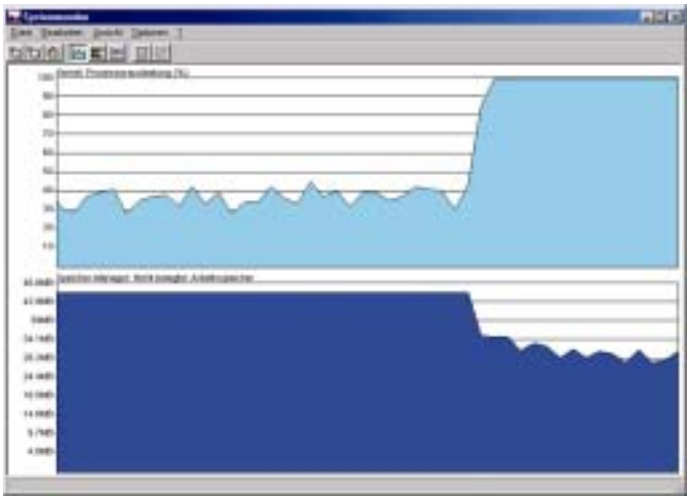

Abbildung 23: Systemarchitektur 1: Ressourcen[A]

Dennoch sind einige Probleme dieser Architektur klar geworden. Diese zeigen sich hauptsächlich auf Clientseite. Wie erwähnt kommt es durch die Kombination von Windows Media Player und Viewpoint Media Player zu unvorhersehbaren Reaktionen. Es scheint, als ob sich die beiden Plug-ins zeitweise gegenseitig die vom Rechner zur Verfügung stehenden Ressourcen abgraben und so oft nichts mehr funktioniert. Das hat eine Analyse der Prozessorlast und des Arbeitsspeichers in solchen Situationen ergeben. Die Prozessorlast steigt in diesen Fällen sprungartig auf bis zu 100% und auch der freie Arbeitsspeicher sinkt merklich. Eine Lösung sind Hilfsfunktionen, wie z.B. `freeMPlayer()`. Der massive Einsatz von JavaScript kann gerade wenn es darauf ankommt Befehle, in sehr kurzen Zeitabständen abzusetzen, auch aufgrund der browserinternen JavaScript Implementierung zu Problemen führen. *„Die aktuelle Implementierung von JavaScript ist nicht besonders schnell. Bei einer komplexeren Berechnung oder Manipulation einer HTML-Seite hat man manchmal den Eindruck, als ob der Rechner eingeschlafen ist.“* [1] – das schreibt auch Laura Lemay in ihrem JavaScript-Buch. Wirklich vorhersehen, ob durch die Integration zusätzlicher Skriptkommandos und Animationen weitere Probleme dieser oder ähnlicher Art auftauchen, kann man allerdings nicht. Dies kann erst die praktische Umsetzung zeigen.

[A] Getestet unter Windows 98, Pentium II, 333Mhz, 192 MB RAM. Über JavaScript Befehle wurde vom Windows Media Player 7 auf das Viewpoint Plug-in zugegriffen. Beide Plug-ins befanden sich innerhalb des Internet Explorers 5.5.

Insgesamt ist das System dennoch als gut und durchaus praktikabel zu beurteilen. Probleme, die während dieser Implementierung auftraten, ließen sich durch Workarounds lösen. Großer Vorteil innerhalb eines Systems, das eine breite Zielgruppe ansprechen soll, ist sicherlich die starke Verbreitung von Windows Media und zukünftig wohl auch Viewpoint. Alle wichtigen Funktionalitäten, wie das Steuern von Sprache und Animation des Avatars, lassen sich realisieren. Einzig die Mundbewegungen durch Animationsschleife wirken bei menschlichen Charakteren evtl. unnatürlich. Das System eignet sich demnach vornehmlich für Phantasiefiguren.

6.2. Architektur 2

Nachdem Architektur 1 erfolgreich umgesetzt wurde soll nun auch die zweite Architektur in ein real existierendes System überführt werden. Da die CharActor Technologie die Verwendung von TTS standardmäßig unterstützt liegt der Schwerpunkt bei dieser Implementierung etwas anders als beim ersten System. Es geht nicht in erster Linie darum, Text in Ton zu überführen und Möglichkeiten zu finden, diesen lippensynchron wiederzugeben. Vielmehr soll KiwiLogic LinguBot in das System integriert und die Bewegung bzw. das Verhalten des Charakters effektiv gesteuert werden. Zur Steuerung des Charakters wird die Programmiersprache CharaScript eingesetzt. Charamel stellt auch hier einige Skripte zur Verfügung, die als Basis für eigene Entwicklungen dienen können. Wie die Skripte zur Charaktersteuerung funktionieren, wo spezielle Anpassungen nötig sind und was speziell zur Anbindung des KiwiLogic LinguBot zu entwickeln ist, soll im Folgenden erläutert werden.

6.2.1. Implementierung

Als Server wird auch bei dieser Entwicklung ein Windows Rechner gewählt. Die Firma Charamel hat für die Entwicklung im Rahmen dieser Arbeit freundlicherweise eine entsprechende Entwicklungsumgebung mit installiertem Elan Speech TTS-System zur Verfügung gestellt.

Die Implementierung besteht aus zwei unterschiedlichen Bereichen, von denen in dieser Arbeit nur einer Thema ist. Zum einen ist das die Entwicklung des 3D-Charakters, sowie dessen Modellierung und Animation mit entsprechender 3D-Software, bzw. einem Motion-Capture-System. Auch der Export der 3D-Daten in eine Szenendatei, welche den Charakter und seine Umgebung enthält, sowie der Export der Bewegungsdaten in Bewegungsdateien, durch ein entsprechendes Plug-in der Firma Charamel für 3d studio max, sind Teile dieser Aufgabe. Diese Bereiche werden hier nicht behandelt. Zum anderen müssen die Dateien in einem Avatar-System zusammengeführt werden, die Steuerung des Charakters muss programmiert und der LinguBot angebunden werden. Dieser Teil ist Thema der nächsten Kapitel.

6.2.1.1. Hauptprogramm

Die Kapitel 6.2.1.1, 6.2.1.2, 6.2.1.3 und 6.2.1.3 können aus Gründen der Geheimhaltung nicht veröffentlicht werden.

Abbildung 24: Avatar: Wiseman – Systemarchitektur 2

6.2.2. Analyse und Bewertung

Bei CharActor handelt es sich um eine Technologie, die zur Entwicklung eines Avatar-Systems ausgesprochen gut geeignet ist. Schon die theoretische Planung hat gezeigt, dass sich die Architektur des Systems sehr übersichtlich darstellt. Viele Funktionalitäten, wie. z.B. TTS oder Phonemanalyse, sind bereits in die Technologie eingearbeitet. Auch die praktische Umsetzung hat die Tauglichkeit von CharActor bestätigt. Während der Programmierung traten keine ungeahnten Hindernisse auf, die Entwicklung gestaltet sich sehr effektiv.

Einziger Nachteil bei der Verwendung des CharActor Systems ist die bislang geringe Verbreitung des Browser Plug-ins. Es bleibt abzuwarten, ob sich die Verbreitungszahlen in Zukunft erhöhen werden. Der Installationsprozess des Plug-ins ist äußerst schnell und komfortabel. Eine Bestätigung reicht aus, um das Plug-in mit den benötigten Komponenten in einem automatisierten Prozess zu installieren. Ein Neustart des Rechners ist nicht notwendig.

Das mit CharActor entwickelte System läuft äußerst stabil. Die ausgefeilte Steuerung von Bewegung und Emotion lassen den Charakter sehr natürlich wirken. Besonders die automatische Ansteuerung von Morphtargets des Mundes zur lippensynchronen Audiowiedergabe machen diese Technologie zu einer sehr guten Wahl bei der Entwicklung eines Avatar-Systems.

[1] Lemay, Laura: JavaScript: Interaktive Web-Seiten mit JavaScript, München 1997, S. 37.

7. Fazit

Das Internet ist im ständigen Wandel. Neue Technologien tauchen auf, vorhandene verschwinden wieder aufgrund technischer Mängel, falscher Marketingstrategien oder einfach aufgrund mangelnder Akzeptanz. So wie es Video und Audio geschafft haben, im Internet Einzug zu halten, ist zu erwarten, dass auch der 3D-Bereich vor diesem Medium nicht halt machen wird und sich Anwendungsfelder zeigen werden, in denen 3D-Technologie sinnvoll eingesetzt werden kann. Avatare sind eines der Einsatzgebiete, denen großer Erfolg prophezeit wird. Noch sind entsprechende Technologien nicht sehr verbreitet, Benutzer sind im Umgang mit diesen neuen Erscheinungen kaum vertraut. Dennoch gibt es Technologien, die für einen 3D-Online-Avatar eingesetzt werden können.

Ziel dieser Diplomarbeit war es, das Thema 3D-Online-Avatar genauer zu durchleuchten. Mögliche Einsatzfelder wie E-Commerce, E-Learning, Content-Providing (News) oder Entertainment wurden genannt, verschiedene Technologien, die prinzipiell in einem Avatar-System eingesetzt werden können, wurden daraufhin analysiert. Zwei unterschiedliche Systemarchitekturen wurden zunächst theoretisch entwickelt und schließlich auch praktisch als real funktionierende Systeme umgesetzt. Auf Stärken und Schwächen jedes einzelnen Systems wurde hingewiesen.

Gesamtergebnis dieser Arbeit ist nun aber vielmehr, dass es auf unterschiedliche Art und Weise, durch Kombination verschiedener Technologien auch heute schon möglich ist, ein Avatar-System zu entwickeln, das professionellen Anforderungen gerecht wird, marktfähig ist und im Internet

eingesetzt werden kann. Sicherlich wird es einige Zeit brauchen, bis derartige Systeme Verbreitung und Akzeptanz finden. Berührungsängste auf Anbieter-, wie auf Benutzerseite sind noch groß. Vorteile und mögliche Einsatzgebiete müssen von beiden Seiten zunächst realisiert werden. Die Schnittstelle zwischen Mensch und Maschine ist hier anders als das Computerbenutzer von anderen Anwendungen gewohnt sind. Mit welchem Programm kann man schon derart kommunizieren?

Das größte Problem eines Avatar-Systems ist nach den gewonnen Erkenntnissen nicht in erster Linie in technischen Mängeln zu suchen. Hier sind praktikable Lösungen erarbeitet worden. Vielmehr wird es noch Zeit brauchen bis die Akzeptanz, der Wunsch und die Nachfrage nach derartigen Systemen steigen wird. Nicole C. Krämer schreibt in einem medienpsychologischen Artikel, dass Schlussfolgerungen bezüglich sozialer Effekte nur als vorläufig gelten können, da sich diese mit der steigenden Selbstverständlichkeit von virtuellen Helfern, die einen vom Bildschirm anblicken, verlieren könnten – *„ähnlich wie Rezipienten nach einer gewissen Gewöhnung an den Nachrichtensprecher nicht mehr vor dem Fernsehabend das Wohnzimmer aufgeräumt oder sich gar frisiert haben.“* [1] Tatsache ist, dass Avatar-Systeme gerade im Internet Leistungen erbringen können, wie sie im Moment vergeblich gesucht werden. Sobald sich der Mensch an eine intensivere Kommunikation mit dem Computer gewöhnt hat, wird es gerade im Internet unterschiedlichste Avatar-Systeme geben, die den Benutzer bei verschiedensten Aufgaben unterstützen.

Abschließend möchte ich mich noch bei meiner Kommilitonin Armella Wagner-Herppich, bei Alexander Reinecke von der Firma Charamel und bei Björn Gülsdorff von KiwiLogic für die Unterstützung und die schöne Zusammenarbeit bedanken. Außerdem bedanke ich mich bei Prof. Dr. Johannes Schaugg und Prof. Uwe Schulz, die diese Arbeit betreuten.

[1] Krämer, Nicole C.: Medienpsychologische Methoden - Können virtuelle Helfer uns wirklich helfen. In: Zeitschrift für Medienpsychologie, 14 (N.F.2) 1, S. 36, 37.

8. Literaturverzeichnis

8.1. Monographien

Alvear, José	Webdeveloper.com – Guide to Streaming Multimedia New York: Wiley Computer Publishing, 1998
Apple Computer, Inc.	QuickTime for the Web – A Hands-On Guide for Webmasters, Site Designers, and HTMLAuthors San Francisco: Morgan Kaufmann, 2000
Becker, Barbara	Künstliche Intelligenz: Konzepte, Systeme, Verheißungen Frankfurt/Main; New York: Campus Verlag 1992
Black, Uyless	Internet-Technologien der Zukunft München: Addison-Wesley-Longman, 1999
Hase, Hans-Lothar	Dynamische Virtuelle Welten mit VRML 2.0 – Einführung, Programme und Referenz Heidelberg: dpunkt Verlag, 1997
Lemay, Laura	JavaScript: Interaktive Web-Seiten mit JavaScript München, Markt & Technik, 1997
Niessner, Andreas	VRML-Praxis: VRML 2.0 – Virtuelle Welten dreidimensional modellieren München: Pflaum, 1997
Phillips, Lee Anne	XML - Modernes Daten- und Dokumentenmanagement München: Markt & Technik, 2002

Towner, George / Apple Computer, Inc. — Discovering QuickTime – An Introduction for Windows and Macintosh Programmers
San Francisco: Morgan Kaufmann, 1999

Vary, Peter / Heute, Ulrich / Hess, Wolfgang — Digitale Sprachsignalverarbeitung
Stuttgart: Teubner, 1998

8.2. Artikel

Koglin, Ilona — Visuell greifbar
In: Page, 03/2002, S. 72-77

Loviscach, Jörn — Bauchredner – Die Animationssoftware TrueSpace 5.2 bringt 3D-Köpfe automatisch zum Sprechen
In: c't 9, 22.04.2002, S.65

Krämer, Nicole C. — Medienpsychologische Methoden - Können virtuelle Helfer uns wirklich helfen? - Verfahren zur Evaluation von anthropomorphen Mensch-Technik-Schnittstellen
In: Zeitschrift für Medienpsychologie, 14 (N.F.2) 1, S. 34-37

Puscher, Frank — Bau dir deinen Avatar
In: Internet World, Februar 2002, S. 84-87

8.3. Internetquellen

Apple Computer, Inc. — Apple's QuickTime 4 Surpasses 100 Million Mark
URL : http://www.apple.com/pr/library/2000/oct/10qtmomentum.html
[Stand: 16.08.2002]

Apple Computer, Inc. — How popular is QuickTime?
URL : http://www.apple.com/quicktime/products/qt/faq.html
[Stand: 16.08.2002]

Apple Computer, Inc.	PlainTalk 1.5.4 Document and Software URL: http://docs.info.apple.com/article.html?artnum=60338&SaveKCWindowURL=http%3A%2F%2Fkbase.info.apple.com%2Fcgi-bin%2FwebObjects%2Fkbase.woa%2Fwa%2FSaveKCToHomePage&searchMode=Expert&kbhost=kbase.info.apple.com&showButton=false&randomValue=100&showSurvey=true&sessionID=anonymous|140466715 [Stand: 14.08.2002]
Bennett , Graeme	MacOS 8.5 - Is Apple thinking differently? URL: http://thetechnozone.com/macbuyersguide/software/system/MacOS85.html [Stand: 14.08.2002]
Elan Speech	Speech Engine URL: http://www.elantts.com/products/fp_speechengine.html [Stand: 26.08.2002]
Heise Online	RealPlayer für UMTS-Handys URL. http://www.heise.de/newsticker/data/jk-28.06.00-006/ [Stand: 18.08.2002]
Heise Online	RealNetworks macht auf Open Source und ärgert Microsoft URL: http://www.heise.de/newsticker/data/vza-23.07.02-000/ [Stand: 18.08.2002]
Heise Online	SIGGRAPH: 3D-Standard fürs Web nimmt Formen an URL: http://www.heise.de/newsticker/result.xhtml?url=/newsticker/data/js-24.07.02-003/default.shtml&words=VRML [Stand: 25.08.2002]
Institut für Kommunikationsforschung und Phonetik der Universität Bonn	HADIFIX URL: http://www.ikp.uni-bonn.de/~tpo/Hadifix.html [Stand: 15.08.2002]
MBROLA Project Development Team, Circuits Theory and Signal Processing Lab, Faculté Polytechnique de Mons	The MBROLA Project - Towards a Freely Available Multilingual Speech Synthesizer URL: http://tcts.fpms.ac.be/synthesis/mbrola/mbrola_entrypage.html [Stand: 15.08.2002]

KiwiLogic	Kiwilogic Lingubot: Das meistverkaufte natürlichsprachliche Dialogsystem der Welt. URL: http://www.kiwilogic.de/kiwilogic/site/_xml/cont_index.php?menue_id=10055&submenue_id&id [Stand: 15.08.2002]
KiwiLogic	KiwiLogic LinguBot Schnittstellen und ihre Einsatzmöglichkeiten URL: http://www.kiwilogic.de/linebreak/mod/netmedia_pdf/data/Lingubot%20Technologie%204.%20Schnittstellen2.pdf [Stand: 01.08.2002]
KiwiLogic	LinguBot Creator und Web Engine - Technische Daten und Systemvoraussetzungen URL: http://www.kiwilogic.de/linebreak/mod/netmedia_pdf/data/Lingubot%20Technologie%203.Technische%20Daten.pdf [Stand: 01.08.2002]
Koordinierungs- und Beratungsstelle der Bundesregierung für Informationstechnik in der Bundesverwaltung im Bundesministerium des Innern	Glossar URL: http://www.bund.de/BundOnline-2005/SAGA/Glossar-.6343.htm [Stand: 29.07.2002]
Microsoft	Windows Media Player: Advanced Scripting for Cross-Browser Functionality URL: http://msdn.microsoft.com/library/en-us/dnwmt/html/cross-browser.asp?frame=true#Understanding%20script%20commands [Stand: 24.08.2002]
Mummert + Partner Unternehmensberatung Aktiengesellschaft	Virtuelle Assistenten: Geheimwaffe gegen Milliardenverluste im Internet URL: http://www.mummert.de/deutsch/press/a_press_info/01030a.html [Stand: 24.07.2002]
Quinn, G. / Wang, H.-P. / Martinez, D. / Bourne, P.E.	Developing Protein Documentaries And Other Multimedia Presentations For Molecular Biology - Update on client-side speech synthesis URL: http://www.sdsc.edu/pb/papers/psb99_footnote.pdf [Stand: 14.08.2002]

RealNetworks	RealNetworks Announces Helix — The First Comprenhensive, Open Standard For Digital Media Delivery URL: http://www.realnetworks.com/company/press/releases/2002/helix_community.html [Stand: 18.08.2002]
RealNetworks	RealProducer Plus® User's Guide URL: http://www.service.real.com/help/library/guides/producerplus85/producer.htm [Stand: 18.08.2002]
RealNetworks	The Helix™ Platform URL: http://www.helixcommunity.org/content/platform.html [Stand: 18.08.2002]
RealNetworks	The Helix™ Vision URL: http://www.helixcommunity.org/content/vision.html [Stand: 18.08.2002]
Robben, Matthias	Service Sells – Kundenberatung online URL: http://www.ecin.de/strategie/kundenberatung/ [Stand: 24.07.2002]
Springer, Uta	Avatar, vertritt mich! Newsletter 12/2002 der comvos online medien GmbH URL: http://www.comvos.de/media/pdf/newsletter1200.pdf [Stand: 31.07.2002]
Tang, Miriam	Virtuelle Figuren sollen das Web “persönlicher” machen URL: http://www.heise.de/newsticker/result.xhtml?url=/newsticker/data/cp-01.07.01-001/default.shtml&words=Avatar [Stand: 01.08.2002]
Towner, George / Apple Computer, Inc.	Discovering QuickTime – An Introduction for Windows and Macintosh Programmers San Francisco: Morgan Kaufmann, 1999
Viewpoint	Creating 3D Rich Media Web Applications URL: http://www.viewpoint.com/developerzone/docs/Create3DApps.pdf [Stand: 27.08.2002]

Viewpoint	Viewpoint Experience Technology: Technical Overview URL: http://www.viewpoint.com/developerzone/docs/VETTechOver.pdf [Stand: 15.06.2002]
Viewpoint	Viewpoint Experience Technology XML Reference Guide URL: http://www.viewpoint.com/developerzone/docs/xmlreference.pdf [Stand: 15.06.2002]
Waller, Richard	How Big is the Internet URL: http://www.waller.co.uk/web.htm [Stand: 16.08.2002]

8.4. Sonstige Quellen

Charamel	CharaScript Reference Vertrauliches PDF-Dokument der Charamel GmbH, Richard-Wagner-Strasse 39, 50674 Köln
Charamel	Einbindung virtueller Charaktere in Webseiten mit CharActor Vertrauliches PDF-Dokument der Charamel GmbH, Richard-Wagner-Strasse 39, 50674 Köln
Glaser, Rob / RealNetworks	Präsentation und Diskussionsrunde zur Markteinführung von Helix am 22. Juli 2002 in San Francisco Aufzeichnung: http://play.rbn.com/?url=tornado/tornado/demand/index_other.smil&proto=rtsp [Stand: 18.08.2002]
Gülsdorff, Björn / KiwiLogic	Gespräch mit Björn Gülsdorff, Leiter des Autoren-Teams von KiwiLogic, Am Sandtorkai 77, 20457 Hamburg am 03.06.2002 E-Mail-Verkehr
Reinecke, Alexander / Charamel	Gespräch mit Alexander Reinecke, CEO Development, Charamel GmbH, Richard-Wagner-Strasse 39, 50674 Köln am 31.05.2002 und 12.08.2002 E-Mail-Verkehr