

Uwe Kloos, Natividad Martínez, Gabriela Tullius (Hrsg.)

Informatics Inside:

**Grenzen überwinden –
Virtualität erweitert Realität**

Informatik-Konferenz an der Hochschule Reutlingen
11. Mai 2011

ISBN 978-3-00-034591-3



Hochschule Reutlingen
Reutlingen University

Impressum

Anschrift

Hochschule Reutlingen
Reutlingen University
Fakultät Informatik
Medien- und Kommunikationsinformatik
Alteburgstraße 150
D-72762 Reutlingen

Telefon: +49 (0)7121 / 271-4002

Telefax: +49 (0)7121 / 271-4042

E-Mail: infoinside@reutlingen-university.de

Internet: <http://www.infoinside.reutlingen-university.de>

Organisationskomitee:

Prof. Dr. Uwe Kloos, Hochschule Reutlingen
Prof. Dr. Natividad Martínez, Hochschule Reutlingen
Prof. Dr. Gabriela Tullius, Hochschule Reutlingen
Dipl.-Betriebswirtin (BA) Danja Walz, Hochschule Reutlingen

Fabian Denninger
Clemens Fiedler
Daniel Grbavac
Christine Heberle
Dirk Höntzsch
Philipp Kallenberg
Fabian Liedtke
Denis Merkle
Tobias Ottmar
Alexander Rometsch
Patrick Schuler
Ralf Schultheiß
Kai Stenkamp
Nicole Stiner
Bernhard Thiersch

Copyright: © Hochschule Reutlingen, Reutlingen 2011
Herstellung und Verlag: Hochschule Reutlingen
ISBN 978-3-00-034591-3



Vorwort

Die Informatics Inside – Konferenz findet in diesem Jahr zum dritten Mal statt. Mit dem Thema Grenzen überwinden – Virtualität erweitert Realität stellt sich die Veranstaltung einem aktuellen Schwerpunkt, der viele Interessierte aus Wirtschaft, Wissenschaft und Forschung anzieht. Die Konferenz hat sich von einer Veranstaltung für die Masterstudenten des Studiengangs Medien- und Kommunikationsinformatik zu einer offenen Studentenkonzferenz entwickelt.

Um die Qualität weiter zu steigern wurde parallel dazu ein zweistufiges Review-Verfahren für Beiträge dieses Tagungsbandes eingeführt. Jeder der begutachteten Beiträge wird durch einen Vortrag vorgestellt und in diesem Tagungsband abgedruckt.

Das Programm der Konferenz besteht in diesem Jahr aus Präsentationen und Ausstellungen. Zahlreiche Firmen und Forschungseinrichtungen sind durch eigene Stände und Vorträge vertreten.

Als Mitglied des Organisationskomitees möchte ich mich besonders bei den teilnehmenden Masterstudenten bedanken, die die Organisation maßgeblich getragen haben und die Arbeit sehr professionell geplant und durchgeführt haben. Darüber hinaus danke ich den Autoren, Firmen, Forschungsinstituten und Partnern für die Unterstützung. Ich wünsche Ihnen eine spannende Konferenz an der Hochschule Reutlingen.

Prof. Dr. Natividad Martínez

Inhaltsverzeichnis

PROGRAMMIERSCHNITTSTELLEN VON AUGMENTED-REALITY-BROWSERN IM RAHMEN DES PROJEKTS „AUGMENTURE“	6
---	----------

PHILIPP KALLENBERG

MOBILE AUGMENTED-REALITY AUF SMARTPHONES MIT DEM AR-BROWSER JUNAIO	11
---	-----------

RALF SCHULTHEIß

ENTWURFSENTSCHEIDUNGEN ZUR AUGMENTED-REALITY-SCHNITZELJAGD „AUGMENTURE“17	
--	--

BERNHARD THIERSCH

DIDAKTISCHE UND SPIELKONZEPTIONELLE ASPEKTE EINER AUGMENTED-REALITY-SCHNITZELJAGD AM BEISPIEL „AUGMENTURE“	22
---	-----------

BENJAMIN ROMER, CHRISTINE HEBERLE

XML3D PHYSIK - SPEZIFIKATION EINER PHYSIKERWEITERUNG UND INTEGRATION EINER PHYSIK-ENGINE IN EINEN XML3D-FÄHIGEN BROWSER	30
--	-----------

FABIAN LIEDTKE

BROWSERBASIERTE MULTITOUCH-INTERAKTIONEN ZUR STEUERUNG VON XML3D SZENEN..	34
--	-----------

DIRK HÖNTZSCH, KAI STENKAMP

KUNDENSPEZIFISCHE ANPASSUNGEN EINES OBJEKTORIENTIERTEN SYSTEMS UNTER DER VERWENDUNG DES SERVICE-LOCATOR-MUSTERS	40
--	-----------

PATRICK SCHULER

EIN PATTERN-KATALOG FÜR REST-KONFORME ARCHITEKTUREN	44
--	-----------

BENJAMIN BLUBACHER, RALPH KELKENBERG, FRANK GRIESINGER, MARITA KLEIN, MICHAEL HAAS,

PROF.DR.ALFRED ZIMMERMANN

EINSATZ VON GRAPHENALGORITHMEN ZUR PLANUNG DES ZUSPIELS IM ROBOTERFUßBALL 49	
---	--

HANNES BERGLER, DANIEL METZGER, URSULA VOß

TIERVERHALTEN AUF DEM ROBOTER ASURO	53
ALEXANDER ROMETSCH	
ANWENDUNGSENTWICKLUNG ZUR UNTERSUCHUNG MOTORISCHEN LERNENS	55
ALEXANDER ROMETSCH	
KÜHLUNG BEI LED-PROJEKTOREN: EIN VERGLEICH VON ZWEI KÜHLSYSTEMEN	58
DANIEL GRBAVAC	
COMPARING DIFFERENT LATENCY-MEASURING METHODS FOR IMAGE TRANSMISSIONS	64
DENIS MERKLE	
VIRTUELLE REALITÄT BEI MEDIEN- UND KOMMUNIKATIONSINFORMATIKERN	70
PATRICK GAßNER, SAMET KESER, JULIAN DOLL, CHRISTOPH MEIBNER, EMIL WIESE, ERIK HERRMANN	
WMC² WEB * MOBILE * CONTENT * COLLABORATION	73
BENJAMIN BLUBACHER, NICO HORNUNG	

Programmierschnittstellen von Augmented-Reality-Browsern im Rahmen des Projekts „Augmenture“

Philipp Kallenberg
Hochschule Reutlingen

philipp.kallenberg@student.reutlingen-university.de

ABSTRACT

Dieser Artikel befasst sich mit Programmierschnittstellen von sogenannten Augmented-Reality-Browsern für mobile Endgeräte. Im Rahmen des Projektes „Augmenture“, einer „Augmented-Reality-Schnitzeljagd“, sollen die Schnittstellen dahingehend geprüft werden, ob sich damit dynamische Inhalte und interaktive Spielabläufe entwickeln lassen. Dazu werden die Schnittstellen zweier Augmented-Reality-Browser (Junao und Layar) analysiert, an den Projektanforderungen gemessen und bewertet. Abschließend wird eine mögliche technische Umsetzung des „Augmentures“ aufgezeigt.

1. EINLEITUNG

1.1 AUGMENTED-REALITY

“AR systems integrate virtual information into a person's physical environment so that he or she will perceive that information as existing in their surroundings.” So definieren Karimi und Hammad [1] die sogenannte „Augmented Reality“, kurz AR. Um die reale Umgebung einer Person und digitale Inhalte verschmelzen zu lassen, muss diese Person die reale Umgebung durch ein Display betrachten, auf dem zusätzlich die digitalen Inhalte abgebildet werden. Zusätzlich sollen nach Azuma [2] Echtzeitinteraktionen mit den digitalen Inhalten und eine perspektivisch korrekte Ausrichtung der Inhalte (Registrierung) gegenüber der realen Umgebung möglich sein. Mehr Information zur technischen Umsetzung der Interaktionsmöglichkeiten und Displays bietet [3], eine tiefer gehende Einführung in Augmented-Reality ist in [4] nachzulesen.

1.2 MOBILE AR-BROWSER

Ein mobiler Browser ist nach [5] „optimiert für die kleinen Bildschirme und die begrenzten Ressourcen eines tragbaren Geräts, wie beispielsweise eines Smartphones.“ Diese Browser stellen üblicherweise Webseiten auf Smartphones dar. AR-Browser bieten darüber hinaus die Möglichkeit digitale Inhalte (Points of Interest, kurz POIs) in Zusammenhang mit der realen Umgebung anzuzeigen. Sie nutzen die Struktur standortbezogener Dienste [6] und stellen AR-Inhalte bereit. Dazu muss das

mobile Endgerät nach [7] folgende Voraussetzung erfüllen:

- Positionsbestimmung (GPS, Kompass)
- Abbildung der Umgebung (Kamera)
- Abrufen der Inhalte (Internetverbindung)

So können AR-Browser dem Nutzer digitale Inhalte standortbezogen bereitstellen und mit dem Kamerabild der realen Umgebung kombiniert darstellen.

1.3 AUGMENTURE

„Augmenture“ ist ein Projekt der Hochschule Reutlingen, in dem eine sogenannte „Augmented-Reality-Schnitzeljagd“ umgesetzt werden soll. Eine herkömmliche Schnitzeljagd soll hier, mit Hilfe von Smartphones und AR-Browsern, um digitale Inhalte erweitert werden. Inhaltliche Hintergründe und Abläufe einer solchen „AR-Schnitzeljagd“ sind in [8] nachzulesen.

2. API-ANFORDERUNGEN

Das zuvor genannte „Augmenture“ soll mit Hilfe eines AR-Browsers umgesetzt werden. Dazu müssen für den AR-Browser POIs erzeugt werden, die der Anwender später zur Durchführung des Augmentures benötigt. Hierfür stellen einige Browserhersteller Programmierschnittstellen (APIs) bereit, die es in unterschiedlichem Umfang erlauben, eigene AR-Inhalte, Interaktionsmöglichkeiten und Logik in die jeweiligen AR-Browser zu integrieren. Im Hinblick auf die Umsetzung des Augmenture-Projekts, werden nachfolgend Anforderungen an die APIs formuliert.

2.1 DYNAMISCHE INHALTE

Die POIs, die auf dem Endgerät des Nutzers angezeigt werden, sollen von Entwicklerseite zur Laufzeit dynamisch verändert werden können, d.h. es muss möglich sein, POIs vor dem Nutzer zu verstecken oder erst unter bestimmten Umständen (bestimmter Ort, bestimmte Zeit) zu aktivieren. Analog muss sich bei einer konventionellen Schnitzeljagd der Spieler von einem Punkt zum nächsten „hängeln“, d.h. erst beim Erreichen eines Punktes, erhält er den Hinweis auf das nächste Teilziel. Medieninhalte, wie Texte, 3D Objekte, Bilder, Audio- und

Videodateien sollen so mit Hilfe der POIs dynamisch bereitgestellt werden. Ein Gegenbeispiel wäre statischer Inhalt, wie etwa alle Tankstellen in der Umgebung auf ein Mal anzuzeigen.

2.2 INTERAKTION

Der Nutzer soll die Möglichkeit haben, mit den POIs zu interagieren. Auf Entwicklerseite muss über die API erkannt werden, dass eine Interaktion stattfindet, um dann die gewünschte Rückmeldung [9] zu geben. Im Rahmen der Schnitzeljagd könnte ein Nutzer beispielsweise eine Route zu einem weiter entfernten Teilziel (POI) anfordern, Aufgaben lösen oder Medien abspielen. Die API muss eine entsprechende Interaktionsschnittstelle bereitstellen (im einfachsten Fall klickt der Nutzer auf den POI und löst damit ein Ereignis auf dem Server aus).

2.3 DATENHALTUNG

Der Status des Nutzers muss während der Durchführung der Schnitzeljagd in einem Datensatz gehalten werden. Unter anderem muss festgehalten werden, welche Teilziele der Nutzer bereits aufgesucht hat, welche POIs er sehen darf und ob er Aufgaben richtig gelöst hat. Der Datensatz muss zur Laufzeit über die API abgerufen sowie manipuliert werden können. Die Datenhaltung ermöglicht, im Zusammenspiel mit dynamischen Inhalten und Logik, den Ablauf zu steuern. Zudem können in der Datenbasis die POIs mit zugehörigen medialen Inhalten verwaltet werden.

2.4 STRUKTUR

Es soll mit Hilfe der APIs eine Struktur geschaffen werden, die leicht um zusätzliche Inhalte erweiterbar ist. Das heißt konkret, dass der logische Ablauf einer Schnitzeljagd sowie die grundlegenden Interaktionen so implementiert werden können, dass POIs und zugehörige Medien/Aufgaben beliebig hinzugefügt werden können, ohne die Struktur ändern zu müssen.

Optionale Anforderungen sind Schnittstellen zur Bilderkennung, die auf zweierlei Weise nützlich wären: zur Navigation im Innenbereich mit LLA Markern¹ oder Tracking von Markern.

Die Nutzung der APIs sollte kostenfrei sein.

3. APIS VON AR-BROWERN

Die Programmierschnittstellen der AR-Browser Junaio und Layar werden nachfolgend näher betrachtet. Diese beiden Produkte wurden aufgrund zweier Aspekte ausgewählt: Benutzerfreundlichkeit und Entwicklerfreundlichkeit [10], d.h. der Browser

soll einerseits gut bedienbar sein und andererseits sollen ausreichend Möglichkeiten für Entwickler bereitgestellt werden. Abbildung 1 veranschaulicht dies in einem zweidimensionalen Diagramm nach [10]. Anschließend werden nur die aus Entwickler-sicht relevanten Aspekte untersucht: die Programmierschnittstellen.

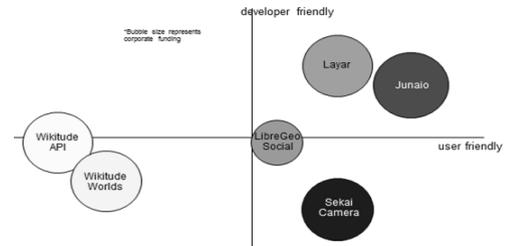


Abbildung 1: AR-Browser-Vergleich nach [10]

3.1 DIE JUNAIO-API

Der Nutzer bekommt hier seine standortbezogenen AR-Inhalte vom zentralen Junaio-Server und muss dazu im Junaio AR-Browser auf dem Smartphone sogenannte „Channels“ abrufen [5]. Der Entwickler hat die Möglichkeit, selbst Channels über ein Web-Frontend zu generieren und diese mit seinen eigenen Inhalten zu füllen. Dabei werden die Channels selbst auf dem Junaio-Server erzeugt, die Inhalte dagegen auf einen externen Server ausgelagert, den der Entwickler bereitstellen muss. Abbildung 2 veranschaulicht die Junaio Architektur nach [11].

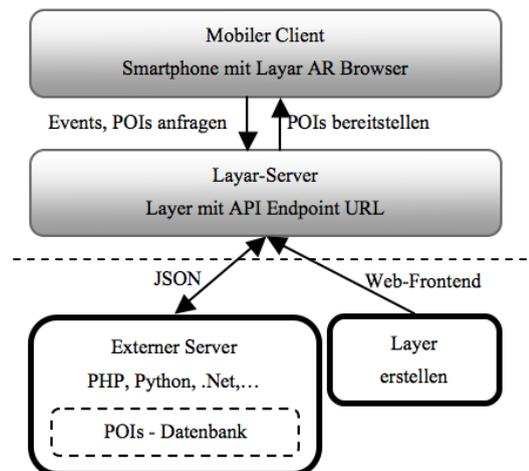


Abbildung 2: Junaios Open API nach [11]

Die POI-Anfrage von Seiten des Clients kann durch verschiedene Ereignisse (Events) ausgelöst werden:

- Ein Channel wird geöffnet.
- Das Endgerät ändert seine Position.

¹ Latitude, Longitude, Altitude in einem Marker kodiert

- Der User interagiert mit POIs.
- Zyklische Aktualisierung von Serverseite. (Junaio-Server)

In der Anfrage werden unter anderem folgende Informationen über den Nutzer transportiert:

- Position und Orientierung des Nutzers
- Eindeutige ID des Nutzers
- Ausgelöstes Ereignis

Der Junaio-Server bearbeitet die Events und ruft die sogenannte Callback-URL auf, um die Anfrage im XML-Format an den externen Entwicklerserver weiterzugeben. Diese Callback-URL muss der Entwickler beim Erstellen seines Channels angeben. Ziel kann ein PHP Skript oder eine ASP.Net Datei sein. Das aufgerufene Skript auf dem Entwicklerserver muss nun eine XML-Datei an den Junaio-Server zurückliefern, welche POIs in vordefiniertem Format² beinhaltet. Anschließend verarbeitet der Junaio-Server die erhaltene XML-Antwort und gibt die aktualisierten Inhalte an den AR-Browser weiter, der diese dann entsprechend darstellt.

Möglichkeiten zur Einflussnahme auf den angezeigten Inhalt hat der Entwickler bei der Erstellung der Skriptdateien. Innerhalb des über die Callback-URL aufgerufenen Skripts, können weitere Skripts aufgerufen werden oder Datenbankzugriffe abgewickelt werden. So kann z.B. die Darstellung bestimmter POIs an Bedingungen geknüpft werden, die im Skript überprüft werden. Durch serverseitige Logik kann eine Art Ablauf implementiert werden, dem der Nutzer folgen muss, ähnlich einem level-basierten Computerspiel.

3.2 LAYAR API

Analog zur Junaio-Struktur des vorherigen Kapitels, fragt der Layar AR-Browser POIs beim Layar-Server an, der diese wiederum vom Entwicklerserver bezieht [12].

Es gibt neben den abweichenden Bezeichnungen – siehe Abbildung 3 (Layer entspricht Channel, API Endpoint-URL entspricht Callback-URL) – zwei Unterschiede zur Junaio Struktur:

- Das Datenaustauschformat JSON anstatt XML
- Die übermittelten Informationen

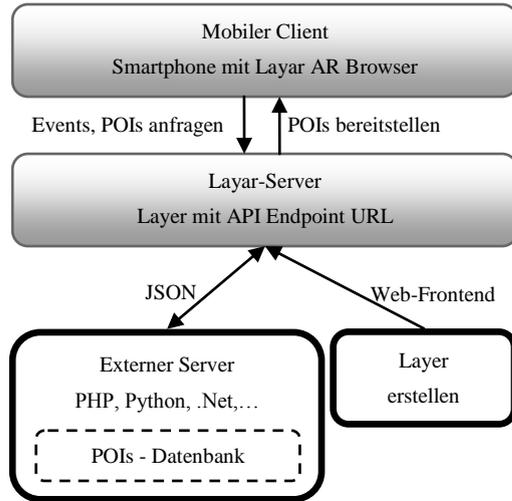


Abbildung 3: Layar Architektur

Die POI-Anfrage des Clients wird hier ebenfalls durch folgende Events ausgelöst:

- Öffnen eine Layers
- Positionsänderung
- Nutzerinteraktion
- Zyklische Aktualisierung von Serverseite (Layar-Server)

Bei der Anfrage werden ebenfalls verschiedene Nutzerinformationen übermittelt:

- Position
- Eindeutige ID
- Ländercode der SIM-Karte
- Spracheinstellung des mobilen Geräts
- Ausgelöstes Ereignis

Auch der Layar-Server bearbeitet die vom Layar-AR-Browser erhaltenen Events und leitet sie mit den obenstehenden Informationen, im JSON-Format³ an den Entwicklerserver weiter. Die Adresse der Datei auf dem Entwicklerserver, die die Anfrage verwerthen soll, muss in der API Endpoint-URL angegeben werden. Inhalte (POIs) müssen in JSON-Objekten verpackt an den Layar-Server zurückgegeben werden. Die serverseitige Entgegennahme der Anfragen und Rückgabe der JSON-Objekte kann in unterschiedlichen Programmiersprachen implementiert werden, z.B. C# oder PHP [13]. Auch hier besteht die Möglichkeit Programmlogik festzulegen darin, Anfragen auf dem Entwick-

²<http://www.junaio.com/publisher/returnparameters>

³JavaScript Object Notion - <http://www.json.org/>

ler-server zu bearbeiten und POIs an Bedingungen geknüpft anzuzeigen.

4. BEWERTUNG

Sowohl die Junaio API als auch die Laya API bieten in ausreichendem Umfang die Möglichkeit, Inhalte dynamisch zu generieren, da der Nutzer in seiner POI-Anfrage Informationen über seine Position sowie seine ID übermittelt. Bis auf die unterschiedlichen Austauschformate ist der Ablauf gleich: Welche POIs dem Nutzer wann angezeigt werden, kann der Entwickler bei der Generierung der Antwort bestimmen. Dort kann er auch die (Medien)Inhalte der POIs festlegen und 3D Modelle, Audio-, Video- und Bilddateien sowie Texte und Links zu externen Webseiten bereitstellen.

Auch die Interaktionsmöglichkeiten sind in beiden Lösungsvarianten ähnlich: Der Nutzer kann auf POIs klicken und sendet damit implizit ein Event an den Server, welches an den Entwicklerserver weitergegeben wird. Der Entwickler kann nun bestimmen, wie er mit diesem Ereignis umgeht. Das Ergebnis eines Events könnte sein, dass der Nutzer auf eine Webseite geleitet wird, auf der er eine Aufgabe löst, es könnte eine Audio-/Videodatei abgespielt werden oder die POIs aktualisiert werden. Dieses rudimentäre Eventhandling, bei dem sich die beiden APIs nur geringfügig unterscheiden, genügt für die eingeschränkten Interaktionsanforderungen des Augmenture-Projekts.

5. ANWENDUNG

Da die Programmierschnittstellen beider AR-Browser die Anforderungen erfüllen und zudem strukturell sehr ähnlich sind, wird eine parallele

Entwicklung für beide Systeme betrieben. Aufbauend auf einer gemeinsamen Datenbasis in Form einer MySQL Datenbank wird ein erweiterbares Grundgerüst für das Augmenture bereitgestellt, das sowohl den Junaio als auch den Laya-Server bedient.

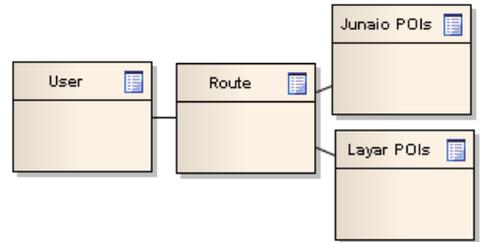


Abbildung 4: Reduziertes Modell der Datenbasis

In der Datenbasis liegen Informationen zum Nutzer, dessen Route und den POIs. Abbildung 4 zeigt ein vereinfachtes Modell der Datenbankstruktur. Dabei kann ein Nutzer einer Route und dieser Route mehrere POIs zugeordnet sein. Zudem ist dem Nutzer ein Level und ein Punktestand zugewiesen [8].

Die Verarbeitung der Anfragen in den jeweiligen Datenaustauschformaten erfolgt in separaten PHP Skripten, genau wie die Bereitstellung der POI Informationen in der Antwort. Welche POIs dem Benutzer zurückgeliefert werden, hängt von dessen Level ab, das beim Erstellen der Antworten in den Skripten aus der Datenbank gelesen und geprüft wird (Logik).

Der in Abbildung 5 dargestellte Augmenture-Server

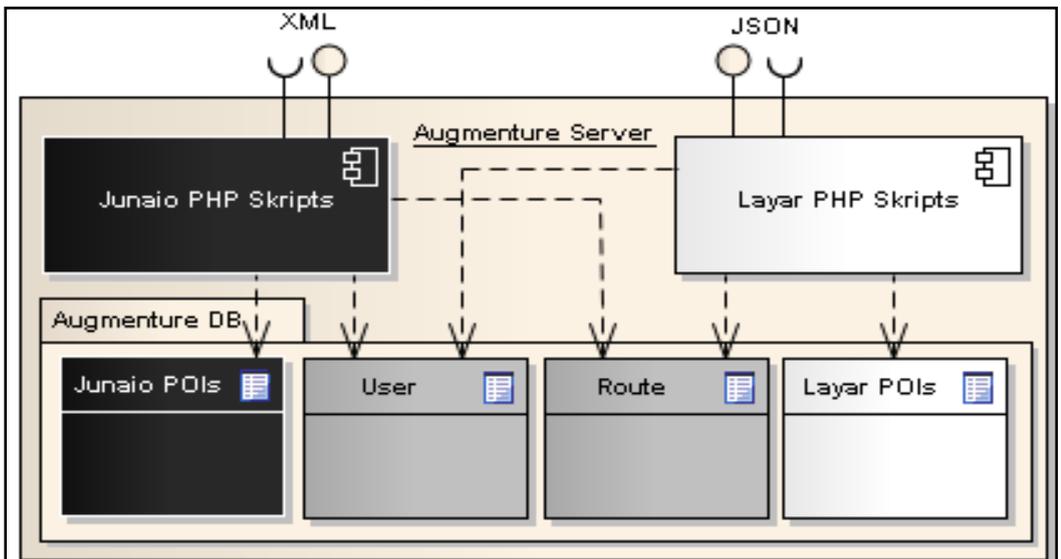


Abbildung 5: Augmenture-Server

ist so zu den Serverschnittstellen beider AR-Browser kompatibel. Abschließend muss auf dem Junaio-Server ein Channel bzw. auf dem Layar-Server ein Layer erstellt werden und das zugehörige Augmenture-Server-Skript in der Callback-URL bzw. im API Endpoint referenziert werden.

Mit dem Augmenture-Server und der hier nicht weiter beschriebenen Logik in den PHP Skripten (Eventhandling, Datenbankabfragen, ...) ist eine Grundstruktur geschaffen, die den Ablauf einer AR-Schnitzeljagd ermöglicht. Neue Inhalte können direkt, ohne weitere Programmierung, in die Datenbank eingefügt und mit beiden AR-Browsern genutzt werden.

6. FAZIT - AUSBLICK

Die von den AR-Browserherstellern angebotenen Programmierschnittstellen ermöglichen die Generierung dynamischer Inhalte in Zusammenspiel mit Benutzerinteraktionen und Programmlogik. So lassen sich Spiele, wie etwa das Augmenture-Projekt, mit diesen Mitteln umsetzen. Kernpunkte sind dabei die Kenntnis über die vom Benutzer ausgeführte Aktion (Eventhandling) sowie die Möglichkeit zur Datenhaltung (Datenbank). Durch die weitgehend sprachunabhängigen Austauschformate XML und JSON kann der Entwickler, solange er sich an die Schnittstellendefinition hält, seine Serveranwendung nach seinen Wünschen gestalten.

Das Bereitstellen von dynamischen Inhalten für AR-Browser setzt allerdings einige Vorkenntnisse in der serverseitigen Programmierung voraus. Derzeit existieren einige Webanwendungen, die nach [5] „eine Integration von Inhalt in den Browser ohne Programmierkenntnisse und eigenen Webspace ermöglichen“. Dabei können aber nur bedingt Interaktionen und Spielabläufe entwickelt und meist nur statische Inhalte erzeugt werden.

Für den Server des Augmenture-Projekts sind neben inhaltlichen auch strukturelle Erweiterungen denkbar: So könnten etwa weitere Datenbanktabellen mit entsprechenden Skripten eingefügt werden, die z.B. ein Monitoring des Augmentures ermöglichen. Der Serverbetreiber erhält zum Zweck der Evaluierung [8] Informationen über den Nutzer, wie etwa Zeit- und Wegverlauf.

Die Programmierschnittstellen der AR-Browser sind in ihrem Umfang für Projekte wie Augmenture ausreichend und bieten genügend Freiheitsgrade, um eine eigene, auf das Projekt zugeschnittene, Serverarchitektur zu entwickeln.

7. LITERATURVERZEICHNIS

- [1] Karimi H. und Hammad A. 2004. Telegeoinformatics: Location-Based Computing and Services. CRC Press 2004.
- [2] Azuma, R. 1997. A Survey of Augmented Reality. Presence: Teleoperators and Virtual Environments 6, 4 (August 1997)
- [3] Zhou, F., Duh, H.B. und Billinghurst, M. 2008. Trends in Augmented Reality Tracking, Interaction and Display: A Review of Ten Years of ISMAR. IEEE International Symposium on Mixed and Augmented Reality 2008.
- [4] Schultheiß, R. 2011. Mobile Augmented Reality auf Smartphones mit dem AR Browser Junaio. Tagungsband zur Informatics Inside 2011. Hochschule Reutlingen.
- [5] Hauser, D. 2011. Konzeption und prototypische Umsetzung einer Rich Internet Application zur Vereinfachung der Integration von Inhalten in einen mobilen Augmented Reality Browser. Master-Thesis. Hochschule Reutlingen.
- [6] Thiersch, B. 2011. Entwurfsentscheidungen zur Augmented Reality Schnitzeljagd „Augmenture“. Tagungsband zur Informatics Inside 2011. Hochschule Reutlingen.
- [7] Michel, J. 2011. Layar & Co.: Apps für Augmented Reality im Überblick. <http://www.teltarif.de/augmented-reality-browser-smartphone-internet/news/42173.html>
- [8] Heberle, C. und Romer, B. 2011. Didaktische und spielkonzeptionelle Aspekte einer Augmented Reality Schnitzeljagd am Beispiel „Augmenture“. Tagungsband zur Informatics Inside 2011. Hochschule Reutlingen.
- [9] Hartmann, F. 2008. Multimedia. Facultas Verlags- und Buchhandels AG.
- [10] Benis, M. 2010. Comparing AR Browsers. <http://mobilegeo.wordpress.com/2010/11/23/comparing-ar-browsers/>
- [11] Metaio. 2010. Junaio for developers –Location Based Channels. http://www.junaio.com/publisherDownload/junaio_LocationBased.pdf.
- [12] Wang, X. 2010. Layar Platform Architecture Overview. <http://layar.pbworks.com/w/page/7783214/Layar-Platform-Architecture-Overview>
- [13] De Smit, J. 2011. PorPOISE for Layar. <http://code.google.com/p/porpoise/>

Letzter Zugriff auf Internetquellen: 18.4.2011

Mobile Augmented-Reality auf Smartphones mit dem AR-Browser Junaio

Ralf Schultheiß
Hochschule Reutlingen

ralf.schultheiss@student.reutlingen-university.de

ABSTRACT

Im Zuge der wissenschaftlichen Vertiefung an der Hochschule Reutlingen wird derzeit eine Schnitzeljagd für potentielle Studienanfängerinnen auf Basis von „Augmented Reality“¹ realisiert. Hierbei sollen mobile Augmented-Reality-Browser zum Einsatz kommen, welche über Smartphones bedient werden können. In dieser Ausarbeitung wird der mobile Augmented-Reality-Browser Junaio hinsichtlich der von Azuma in [1] aufgestellten Kriterien der Augmented-Reality hin untersucht. Nach einer Hinführung zum Thema wird die Augmented-Reality im Allgemeinen und deren Komponenten (Tracking, Präsentation und Interaktion) im Speziellen erklärt. Dabei wird insbesondere auf den AR-Browser Junaio Bezug genommen.

1. HINFÜHRUNG

Eine allgemein gültige Definition des Begriffs „mobiler Augmented-Reality-Browser“ gibt es momentan noch nicht, weswegen Hauser in [2] einen mobilen AR-Browser über diverse Eigenschaften definiert. Eine dieser Eigenschaften ist, dass Augmented-Reality für die Darstellung von Inhalten eingesetzt wird; eine andere, dass er für Smartphones „hinsichtlich des Bildschirms und der zur Verfügung stehenden Ressourcen“ [2] optimiert ist. Nach Wagner [3] handelt es sich bei Handheld-Augmented-Reality um ein AR-System, bei dem der Benutzer das mobile Gerät, auf dem die AR-Anwendung läuft, aktiv in der Hand hält. Demnach kann die Augmented-Reality mit Smartphones als ein Teilgebiet der Handheld-Augmented-Reality angesehen werden.

2. AR-BROWSER JUNAIO

Der mobile AR-Browser Junaio der Firma Metaio Inc. ist eingebettet in einer Client-Server-Architektur. Das Smartphone (mobiler Client) kommuniziert mit dem Junaio-Server, an welchem wiederum externe Server angebunden sind. Die bereitgestellte Programmierschnittstelle (API) erlaubt eine Integration von virtuellen Informationen, Interaktionsmöglichkeiten und Logik. In [4] wird detailliert auf die API von Junaio eingegangen. Die Inhalte („Points of Interest“, kurz POI) werden in sogenannten „Channels“ abgerufen.

Dabei ist es nicht möglich unterschiedliche (ortsbezogene und objektspezifische) Inhaltstypen in einem einzigen Channel zu integrieren [2]. Laut Hersteller kann derzeit Junaio auf Apples iPhone ($\geq 3GS$), iPad, iPod (>3te Generation) und auf Smartphones mit dem Android Betriebssystem (≥ 2.1) eingesetzt werden.

3. AUGMENTED-REALITY

Der Benutzer in einer virtuellen Umgebung ist vollständig von einer synthetischen virtuellen Realität umgeben und kann die reale Welt um sich herum nicht sehen. In der erweiterten Realität hingegen sieht der Benutzer die reale Welt, welche mit virtuellen Objekten überlagert, zusammengesetzt bzw. angereichert ist. [1]



Abbildung 1: Realitäts-Virtualitätskontinuum nach [5]

„Mixed Reality“² ist ein Überbegriff der Mischformen der realen (physischen) nicht modellierten Welt und der vollständig computergenerierten Realität bzw. virtuellen Realität (VR). Eine derartige Umgebung wird entweder von der einen oder der anderen Umgebung mehr dominiert. Das heißt, dass es zwischen den beiden Extremen, reale und virtuelle Umgebung, stufenlos Mischformen und Modellierungsgrade gibt. Augmented-Reality und Augmented-Virtuality sind spezielle Ausprägungen solcher Mischformen. Abbildung 1 verdeutlicht diesen Sachverhalt im Realitäts-Virtualitätskontinuum nach Milgram et al. [5].

Azuma definiert in [1] AR als Systeme mit folgenden Merkmalen:

- ist registriert in 3D
- kombiniert Reales und Virtuelles
- ist interaktiv in Echtzeit

Diese Kriterien machen eine Abgrenzung gegenüber anderen Bereichen virtueller Umgebungen

¹ erweiterte Realität; Abkürzung „AR“

² vermischte Realität

möglich. Somit unterscheidet sich die AR zu der VR besonders durch die Kombination von realer und virtueller Umgebung. Zudem muss eine Registrierung in 3D erfolgen, also ein Bezug der virtuellen Elemente im dreidimensionalen realen Raum mit fester Position und Orientierung geschaffen werden, so dass sie im Idealfall als Einheit und nicht als zwei getrennte Welten wahrgenommen werden. Nach Oehme [6] ist Augmented-Reality eine Anreicherung der realen Welt mit Zusatzinformationen in Form von virtuellen Daten, welche in Abhängigkeit zu der betrachteten Realität situationsgerecht dargestellt werden.

Idealerweise geht es nicht nur darum, die Realität mit virtuellen Objekten anzureichern bzw. hinzuzufügen, sondern auch reale Objekte zu entfernen.

Die virtuelle Realität beschreibt technische Systeme, „in denen die menschlichen Wahrnehmungskanäle weitest möglich durch computergesteuerte Reize beliefert werden“, während in der Augmented-Reality Systeme beschrieben werden, in welchen „Sinnesreize aus der realen Umgebung durch computergesteuerte Daten kontextsensitiv erweitert werden“. [7]

Azuma weist in [1] ausdrücklich darauf hin, dass die oben aufgeführten Charakteristika keine bestimmte Technologie voraussetzen, wie bspw. Datenbrillen oder Displays im Allgemeinen. Reduziert man die physische Realität auf die Summe der menschlichen Sinneswahrnehmungen, ist zwar die visuelle Wahrnehmung des Menschen die am stärksten ausgeprägte, aber dennoch nicht die einzige. Folglich können nicht nur visuelle 3D-Objekte in AR verwendet werden, sondern es kann stattdessen jeder Sinneskanal einzeln oder in Kombination angesprochen werden. Es ist demnach festzuhalten, dass sich die Visualisierung bzw. Darstellung im Sinne der AR nicht auf die visuelle Sinneswahrnehmung beschränkt, sondern als Überbegriff für die Darstellungsformen der verschiedenen Wahrnehmungssinne gesehen werden muss.

4. KOMPONENTEN

Mit der von Azuma in [1] aufgestellten Charakteristika lassen sich drei Kernfunktionalitäten bzw. -komponenten eines AR-Systems ableiten [8]:

- **Tracking und Registrierung:** eine Registrierung in 3D – siehe a) oben
- **Präsentation:** Kombination von Realem und Virtuellem – siehe b) oben
- **Interaktion:** Interaktivität in Echtzeit – siehe c) oben

Nachfolgend werden diese drei Kernkomponenten näher erläutert und in Bezug auf den mobilen Augmented-Reality-Browser Junaio hin untersucht.

4.1 TRACKING

Für eine präzise Überlagerung von virtuellen Informationen in der realen Umgebung ist eine möglichst exakte Erfassung der Position und der Blickrichtung des Anwenders nötig. Der Prozess der Lagebestimmung wird als Tracking bezeichnet. Das lokale Koordinatensystem des Anwenders muss in das Referenzkoordinatensystem des Trackingsystems und weiter in das Koordinatensystem des Renderers transformiert werden. Diesen Prozess nennt man Registrierung. Um die Koordinatensysteme der beiden Welten (real und virtuell) überlagern zu können, muss das Blickfeld des Anwenders bestimmt werden. Hierzu gibt es spezielle Sensoren, die im Stande sind die Position und die Orientierung bezüglich dem Standpunkt und der Blickrichtung des Anwenders zu erfassen. Bei einer Interaktion von realen mit virtuellen Objekten muss zudem auch die Position und Orientierung der realen Objekte bekannt sein. Die Anforderung an ein ideales Tracking fordert präzise, hochauflösende, weitreichende und in Echtzeit gelieferte Daten. Zudem sollte eine uneingeschränkte Bewegungsfreiheit von 6 DOF³ (3D Position und Orientierung) gemessen werden können.

Es gibt eine Vielzahl von Trackingverfahren, die auf unterschiedlichen physikalischen Gegebenheiten basieren. In [9] werden verschiedene Trackingverfahren näher untersucht. Auf aktuellen Smartphones sind Sensoren und MOEMS⁴ vorhanden, womit nachfolgende Trackingverfahren angewendet werden können.

4.1.1 LAUFZEITBASIERT MIT GPS

Beim laufzeitbasierten Trackingverfahren mittels dem „Global Positioning System“ (kurz GPS) ist es möglich, die Position eines Smartphones zu ermitteln. Dabei erfolgt die dreidimensionale Positionsbestimmung mittels eines Empfängers und mindestens vier in Reichweite liegenden, der insgesamt 24 vorhandenen Satelliten. Ein Vorteil dieses Trackingverfahrens ist die hohe Reichweite im Hinblick auf die Nutzung. Jedoch ist der Einsatz in Gebäuden oder bei schlechter „Sicht“ zu den Satelliten nicht möglich. Es können Genauigkeiten bis zu wenigen Metern erreicht werden. Mittels „Differential GPS“ ist eine Erhöhung der Messgenauigkeit möglich. Durch die zusätzliche Nutzung eines elektronischen Kompasses kann zudem eine Aussage über die Blickrichtung gemacht werden. In

³ Abk. für Degrees of Freedom (Freiheitsgrade)

⁴ Abk. für Micro-opto-electromechanical systems

weiterer Kombination mit einem Kreiselinstrument (Gyroskop) kann daneben auch die Orientierungslage des Smartphones bestimmt werden. Die Objekte können mit Hilfe von hinterlegten Koordinaten, Angaben zur Orientierung und zur Größe entsprechend positioniert werden.

Junaio unterstützt das Trackingverfahren mittels GPS. Sind zudem Kompass und/oder Lagesensoren auf dem Smartphone vorhanden, werden diese mit einbezogen.

4.1.2 INERTIALES TRACKING

Inertiales Tracking beruht auf dem Prinzip der Trägheitsnavigation. Die Hauptkomponenten bestehen aus einem Gyroskop und einem Beschleunigungssensor (Accelerometer). Der Beschleunigungssensor registriert eine Bewegung in eine beliebige Richtung. Die Richtung kann durch das Gyroskop bestimmt werden. Hierzu kann auch ein elektronischer Kompass genutzt werden, welcher allerdings nur die Richtung, nicht aber die Orientierung messen kann. Beim Einsatz von Beschleunigungsmesser und Gyroskop ist es nach vorheriger Initialisierung möglich, den momentanen Standort und die Orientierung zu ermitteln. Da die Sensoren über eine gewisse Fehlerrate verfügen, kommt es mit der Zeit zu Abweichungen („Drift“). Deswegen sollte beim inertialen Tracking von Zeit zu Zeit eine Rekalibrierung stattfinden.

Derzeit wird laut [10] zumindest innerhalb des Unifeye-SDK der Firma Metaio inertiales Tracking mittels Beschleunigungsmesser und Kompass auf Apples iPhone und Android Smartphones unterstützt. Da entsprechende Quellen fehlen, kann eine differenzierte Aussage bezüglich des Junaio nicht gemacht werden.

4.1.3 BILDBASIERTES TRACKING

Da die Kamera am Smartphone selbst befestigt ist und die Umgebung beobachtet, spricht man hier von Inside-Out-Tracking. Der Benutzer führt also die aktiven Komponenten des Trackingsystems mit sich. Die Kamera liefert Videobilder, welche von Bildbearbeitungsalgorithmen bearbeitet werden, um bestimmte Merkmale aus den Bildern zu extrahieren (siehe Abbildung 2). Anhand dieser Merkmale kann auf die Position und Ausrichtung der Kamera und/oder des Objektes geschlossen werden. Man unterscheidet zwischen markerbasierten und markerlosen (featurebasierten) Trackingverfahren. Featurebasierte Trackingverfahren verzichten auf künstliche Marker und versuchen stattdessen eine Identifizierung aufgrund natürlicher Strukturen (z.B. Kanten, Ebenen, etc.). Bei markerbasierten Verfahren werden optische Marker generiert und im Raum angebracht. Entweder ist die Position dieser Marker bekannt oder sie fungieren selbst als Nullpunkt. Je nach Aufgabenstellung können diese

Marker unterschiedliche Formen haben. Meist werden solche Marker in Form von zweidimensionalen Bildcodes realisiert. Je flacher der Winkel und je größer die Entfernung, desto schwieriger wird es für die Bildverarbeitung, die Bildmerkmale des Markers zu erkennen. Eine teilweise Verdeckung des Markers führt ebenfalls häufig dazu, dass dieser nicht erkannt werden kann.

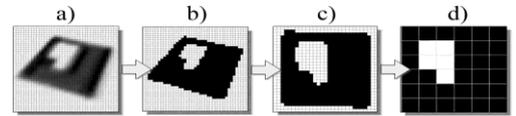


Abbildung 4: Analyse des Videobildes nach [11]:

- a) Erkennen einer Region
- b) Konvertierung
- c) Koordinatentransformation
- d) Skalierung bestimmen

Junaios Bildererkennung geht einen Schritt weiter und kann neben künstlichen Markern zudem ein beliebiges Bild als Marker erkennen. Dazu wird das Bild zuvor auf dem Junaio-Server gespeichert. Dieses wird dann im späteren Vorgang mit den gelieferten Bildern der Smartphonekamera verglichen. Diese Technik⁵ kann im weitesten Sinne als featurebasiertes Tracking angesehen werden.

Eine weitere Besonderheit weist Junaio mit der Verwendung von sogenannten LLA⁶-Markern auf. In diesen zweidimensionalen Bildcodes sind Längen- und Breitengrad, sowie die Meereshöhe kodiert. Wird ein solcher Marker von Junaio erkannt, werden die kodierten Werte verwendet, während die Daten des GPS-Sensors ignoriert werden. Dies ist insbesondere innerhalb von Gebäuden nützlich, da hier GPS, wie bereits angesprochen, normalerweise nicht angewendet werden kann.

4.2 PRÄSENTATION

Wie eingangs festgestellt, sind virtuelle Informationen, welche die Realität anreichern, nicht an bestimmte Technologien gebunden, d.h. dass es keine Festlegung gibt, welche Wahrnehmungskanäle dabei angesprochen werden müssen. Abgesehen von einigen auditiven AR-Systemen spielten bei bisherigen Realisierungen von AR-Systemen visuelle Darstellungstechniken die größte Rolle. Da momentane Smartphones nur visuelle und auditive

⁵ Die Technik wurde an der ETH Zürich (CH) und K.U. Leuven (B) [12] in Zusammenarbeit mit Toyota Motor Corporation entwickelt und wird von der Firma Kooaba AG (CH) vermarktet.

⁶ Abkürzung für Longitude (Längengrad), Latitude (Breitengrad), Altitude (Meereshöhe)

Präsentationen zulassen, werden diese nachfolgend genauer betrachtet⁷.

4.2.1 VISUELL

Bei der visuellen Präsentation von AR sind bestimmte Technologien anzutreffen und lassen sich wie in [13] klassifizieren. Auf aktuellen marktüblichen Smartphones werden unter Ausschluss von Peripheriegeräten meist mobile handbasierte monoskopische Video-See-Through (kurz VST) Displays verwendet. Junaio ist für diese Art von Display optimiert. Bei VST-Displays wird die reale Umgebung zuvor abhängig vom Blickfeld des Benutzers über eine Videokamera gefilmt und auf dem Display wiedergegeben. Eine Transparenz ist im Gegensatz zu den Optical-See-Through (kurz OST) Displays nicht vorhanden. Die Wiedergabe des Kamerabildes kann eine niedrigere Auflösung des Realbildes zu Folge haben und es können je nach eingesetzter Hardware Verzerrungen auftreten. Die Auflösung der aufzeichnenden Kamera und des darstellenden Displays sind geräteabhängig. Da die reale Welt zuvor gefilmt wird, kann es bei der Berechnung des AR-Bildes zu einer Zeitverzögerung bezüglich der eingeblendeten virtuellen Objekte kommen. Dieser Effekt kann aber nivelliert werden, indem die Wiedergabe des Kamerabildes um die Zeit verzögert wird, die das Tracking und die Verarbeitung der Daten benötigen. Zudem können die Kameras nur einen bestimmten Bildwinkel aufnehmen, welcher sich dann bei der Darstellung auf das Sichtfeld (FOV⁸) des Benutzers auswirkt. Da dieses eingeschränkte Sichtfeld unter Umständen POI der Umgebung nicht anzeigt, wird von Junaio ein wie in [14] beschriebenes Visualisierungskonzept (InfoRadar) eingesetzt. Das InfoRadar ist immer egozentrisch, d.h. die Mitte des Radars bildet der Benutzer mit seinem Smartphone. Umliegende POI werden als Punkte entsprechend ihrer Lage dargestellt (siehe Abbildung 3 rechts).

Je nach Anforderung an das AR-System ist ein mehr oder weniger visueller Realismus erforderlich. Wie auch in der VR werden bei der AR meist szenengraphbasierte Renderer eingesetzt. Im Renderingprozess ist bei einer möglichst realistischen Darstellung auf Beleuchtung, korrekte Darstellung von Verdeckungen, Schattenwurf und weiteren Merkmalen innerhalb der realen und der virtuellen Welt zu achten. Für eine Darstellung von 3D-Modellen, wie in Abbildung 3 mittig zu sehen, werden in Junaio die Formate MD2 und Wavefront Objekt unterstützt. Letzteres ist für hochwertige

statische Objekte gedacht. Dagegen kann das MD2-Format für animierte Modelle benutzt werden. Neben einer beschränkten Polygonanzahl, werden in beiden Modellformaten keine Lichtquellen unterstützt. Stattdessen muss die Beleuchtung in die Texturen „gebacken“ werden. Als Schattierungsverfahren wird „Flat Shading“ unterstützt, was sich vor allem bei Objekten mit gekrümmten Flächen negativ in Bezug auf den Realismus auswirken kann.

Neben 3D-Modellen können visuelle Informationsinhalte auch in Form von Videos dargestellt werden. Junaio unterstützt hierbei das MP4-Format. Des Weiteren können Texte und Bilddateien angezeigt werden. Diese POI werden in Form eines 2D-Labels dargestellt (siehe Abbildung 3 links).



Abbildung 5: links: POI als 2D-Label; mittig: Darstellung eines 3D-Modells; rechts: InfoRadar

Billinghurst unterscheidet in [15] drei Modi, wie visuelle Informationen in Augmented-Reality dem Benutzer präsentiert werden können:

Head-stabilized: Die dargestellte Information ist im Sichtbereich des Benutzers fest verankert und ändert sich selbst dann nicht wenn der Benutzer seine Position oder Orientierung verändert.

Body-stabilized: Die Information besitzt eine relativ zum Körper des Benutzers fixe Position. Dabei verändert sich die Darstellung nur, wenn der Benutzer seine Orientierung verändert. Die Position wird nicht berücksichtigt.

World-stabilized: Die Position und die Orientierung der virtuellen Objekte sind mit der realen Umgebung fest verbunden. Die Informationsobjekte ändern sich mit der Orientierung und Position des Benutzers, d.h. die Objekte sind in der realen Welt registriert.

Wie in Kapitel 3 beschrieben, ist nach Azuma die Registrierung der virtuellen Objekte mit der realen Umgebung eine grundlegende Eigenschaft von AR-Systemen, doch lässt sich daraus nicht schließen, dass die Registrierung der Objekte immer World-stabilized sein muss, d.h. auch Mischformen können möglich sein. [8]

Nach Metaio [16] werden aus Sicht der Usability die Distanzen der Modelle in Junaio nach einem bestimmten Algorithmus angepasst. Zudem werden die POI so angeordnet, dass sie sich möglichst nicht überdecken. Dennoch kann durch die Angabe des

⁷ Die Vibrationsmöglichkeit mancher Smartphones wird nicht als haptische Präsentation im Sinne der AR gewertet.

⁸ Abkürzung für „Field of View“ (Sichtfeld)

Parameters „showcorrectperspective“ die korrekte Perspektive erzwungen werden. Die POI werden dann World-stabilized angezeigt. Durch den Parameter „relativetoscreen“ können die POI Head-stabilized dargestellt werden. Ob Darstellungen im Body-stabilized Modus möglich sind, konnte nicht ermittelt werden.

4.2.2 AUDITIV

In Junaio können Audiodateien als stereofonische MP3 (MPEG-1 Audio Layer 3) bis zu 48 kHz abgespielt werden. Unter bestimmten Voraussetzungen lässt sich daraus zwar ein räumliches Klangbild erzeugen, aber eine dynamische Registrierung in Bezug auf die sich ändernde Position und Orientierung des Benutzers bzw. des Objekts in Echtzeit kann mit Junaio nicht realisiert werden. Einen Ansatz für realistische Audiowiedergabe in AR-Systemen liefern Sodnik und Tomazic et al. in [17].

4.3 INTERAKTION

Interaktion ist die Hinzunahme eines Rückkanals, welche eine gezielte Veränderung bzw. Beeinflussung von dargestellten Informationen zulässt. Gutes Interaktionsdesign kann mit entsprechender Usability erreicht werden. Nach Nielsen [18] sollten fünf Ziele für eine gute Usability angestrebt werden: Effektivität, Effizienz, Fehlertoleranz, Lernbarkeit und Sympathie. Im Zusammenhang der Mensch-Maschine-Interaktion sollte der Computer nicht nur spezifische Probleme lösen, sondern dem Benutzer in seinen kognitiven und körperlichen, aber auch in seinen sozialen, emotionalen und kulturellen Eigenschaften assistieren [19]. In dieser Ausarbeitung wurden keine Evaluierungen in Bezug der Usability vorgenommen. In [20] wird auf die Usability von Smartphones und in [21] speziell auf Usability-Aspekte der AR eingegangen.

Bei der AR kann man zum einen auf zahlreiche Interaktionsmetaphern von immersiven virtuellen Umgebungen zurückgreifen. Zum anderen können etablierte zweidimensionale Interaktionskonzepte für graphische Benutzerschnittstellen (GUI), wie bspw. WIMP verwendet werden. Die GUI kann durch Anbindung natürlicher und erkenntnisbasierter Eingaben wie beispielsweise Körperbewegungen, Mimik, Gestik und Sprache ergänzt bzw. ersetzt werden. Oviatt spricht in [22] von multimodalen Schnittstellen (MUI). Nach Azumas Definition muss zudem die Interaktion in Echtzeit möglich sein.

Betrachtet man den Idealfall einer Interaktion in einer Augmented-Reality-Umgebung, so „sollte für den Anwender kein Unterschied in der Interaktion mit realen und virtuellen Objekten zu erkennen sein, reale und virtuelle Szene würden auch auf der

Ebene der Interaktion eine perfekte Synthese eingehen“ [8]. An dieser Idealvorstellung wird in der Forschung gearbeitet, sie ist aber für den praktischen Einsatz, insbesondere bei den momentan verfügbaren Smartphones erst ansatzweise möglich.

Junaio unterscheidet zwischen zwei Interaktionsmöglichkeiten: klickbasierte und hitbasierte Interaktionen. Diese Interaktionen können Ereignisse auslösen, wie das Abspielen von Video- oder Audiodateien. Des Weiteren kann eine hinterlegte URL in einem Webbrowser geöffnet werden. Je nach Nutzung der API können Objekte verändert, entfernt oder hinzugefügt werden. Klickbasierte Interaktionen werden, wie der Name bereits vermuten lässt, durch einen Klick auf das Objekt ausgelöst. Hitbasierte Aktionen werden ausgelöst, wenn sich der Benutzer an einem bestimmten Ortspunkt befindet oder sich in einem bestimmten Radius um einen Ortspunkt aufhält. Bei der Interaktion mit den 2D-Labels setzt Junaio auf das WIMP-Konzept. Eine Manipulation der 3D-Objekte mit Gesten über das Multi-Touch-Display des Smartphones wie in [23] ist nicht möglich. Ebenfalls werden multimodale Schnittstellen nicht unterstützt.

5. ZUSAMMENFASSUNG

Es wurde gezeigt, dass alle drei Komponenten eines AR-Systems im weitesten Sinne in Junaio umgesetzt wurden. Im Tracking werden neben GPS auch bildbasierte Verfahren eingesetzt. Über die LLA-Marker ist zudem ein Indoor-Tracking ohne GPS möglich. Eine Kombination von LLA-Markern und inertialem Tracking wäre wünschenswert. Die virtuellen Informationen können visuell über 3D-Modelle und darüber hinaus als 2D-Label Head- und World-stabilized dargestellt werden. Eine auditive Darstellung ist nur bedingt gegeben. Eine Interaktion mit den dargestellten Informationen kann nur klick- oder hitbasiert erfolgen. Diese Gegebenheiten können je nach Anforderungen für eine Augmented-Reality-Anwendung ausreichen, sind aber natürlich verbesserungsfähig.

Betrachtet man ein ideales AR-System, indem ein hochpräzises Tracking erfolgt, die Präsentation der virtuellen Objekte eine Einheit mit der realen Welt bildet und eine intuitive Interaktion vorherrscht, ist man davon freilich noch weit entfernt.

Echtzeitansätze in Präsentation, Tracking und Interaktion sind von großer Bedeutung, um die Latenzzeit von 70ms zu unterschreiten, da diese Zeit für den kognitiven Arbeitszyklus benötigt wird [13]. Je präziser und ausgefeilter die Techniken in den Einzelkomponenten der Augmented-Reality werden, desto höher wird die Immersion.

Die Entwicklungen von mobilen Endgeräten, sowie die Forschungen in den AR relevanten Gebieten schreiten stetig voran. Deshalb kann man auf die

Weiterentwicklung des mobilen AR-Browsers Junaio im Speziellen und auf andere Augmented-Reality-Systeme im Allgemeinen gespannt sein.

6. LITERATURVERZEICHNIS

- [1] Azuma, R., 1997. A Survey of Augmented Reality, Presence: Teleoperators and Virtual Environments. vol. 6, no. 4, pp. 355-385
- [2] Hauser, D., 2011. Konzeption und prototypische Umsetzung Konzeption und prototypische Umsetzung einer Rich Internet Application zur Vereinfachung der Integration von Inhalten in einen mobilen Augmented Reality Browser. Masterthesis, Hochschule Reutlingen
- [3] Wagner, D., 2007. "Handheld Augmented Reality", Dissertation, Graz University of Technology, Institute for Computer Graphics and Vision
- [4] Kallenberg, P. 2011. Programmierschnittstellen von Augmented Reality Browsern im Rahmen des Projekts Augmenture. Tagungsband II2011. Hochschule Reutlingen
- [5] Milgram, P., Takemura, H., Utsumi, A., Kishino, F. 1994. Augmented Reality: A class of displays on the reality virtuality continuum, SPIE Vol. 2351, In Proceedings of Telemanipulator and Telepresence Technologies
- [6] Oehme, O., 2004. Ergonomische Untersuchung von kopfbasierten Displays für Anwendungen der erweiterten Realität in Produktion und Service, Shaker Verlag
- [7] Heiden, W., Herpers, R., und Hinkenjann, A. 2009. Visual Computing an der FH Bonn-Rhein-Sieg. Forschungsspitzen und Spitzenforschung. Physica-Verlag HD.
- [8] Weidenhausen, J.-M., 2006. Mobile Mixed Reality Platform, Dissertation, Technische Universität Darmstadt
- [9] Rolland, J. P.; Davis, L. D.; Baillot, Y., 2001. A Survey of Tracking Technology for Virtual Environments. In: Barfield, W.; Caudell, T. (Hrsg.): Fundamentals of Wearable Computers and Augmented Reality. Mahwah: Lawrence Erlbaum Associates, Inc., S. 67-112.
- [10] metaio Inc. 2011. Unifeye Mobile Tracking-Configuration. <http://docs.metaio.com/bin/view/Main/Unifeye> Letzter Zugriff 01.04.2011
- [11] Tegtmeier, A., 2006. Augmented Reality als Anwendungstechnologie in der Automobilindustrie. Dissertation. Otto-von-Guericke-Universität Magdeburg
- [12] Quack, T., Bay, H., Van Gool, L. 2008. Object recognition for the internet of things. In Proceedings of the 1st international conference on The internet of things (IOT'08). Springer-Verlag, Berlin, Heidelberg, 230-246
- [13] Reif, R., 2009. Entwicklung und Evaluierung eines Augmented Reality unterstützten Kommissioniersystems, Dissertation, Technischen Universität München
- [14] Rantanen, M., Oulasvirta, A., Blom, J., Tiitta, S., Mäntylä, M., 2004. InfoRadar: group and public messaging in the mobile context. In Proceedings of the third Nordic conference on Human-computer interaction (NordiCHI '04). ACM, New York, NY, USA, 131-140.
- [15] Billinghurst, M., Bowskill, J., Dyer, N., Morphet, J., 1998. An Evaluation of Wearable Information Spaces. In Proceedings of the Virtual Reality Annual International Symposium (VRAIS '98). IEEE Computer Society, Washington, DC, USA
- [16] metaio Inc. 2011. Augmented Reality - junaio... your mobile companion. <http://www.junaio.com> Letzter Zugriff 01.04.2011
- [17] Sodnik, J., Tomazic, S., Grasset, R., Duenser, A., Billinghurst, M., 2006. Spatial sound localization in an augmented reality environment. ACM, 111-118.
- [18] Nielsen, J., 1994. Usability Engineering. San Francisco: Morgan Kaufmann.
- [19] Zhou, F., Been-Lirn Duh, H., Billinghurst, M., 2008. Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR. IEEE Computer Society
- [20] Schneidermeier, T. 2009. Usability of Mobile Devices, Untersuchungen zur Gebrauchstauglichkeit mobiler Endgeräte am Beispiel Apple iPhone, Magisterarbeit, Universität Regensburg
- [21] Fjeld, M., 2004. Usability and collaborative aspects of augmented reality. interactions 11, 6 (November 2004), 11-15
- [22] Oviatt, S., 2007. Multimodal Interfaces. In: The Human-Computer interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications, L. Erlbaum Associates, Hillsdale.
- [23] Hancock, M. Carpendale, S., Cockburn, A. 2007. Shallow-depth 3d interaction: design and evaluation of one-, two- and three-touch techniques. In Proceedings of the SIGCHI conference on Human factors in computing systems (CHI '07). ACM, 1147-1156.

Entwurfsentscheidungen zur Augmented-Reality-Schnitzeljagd „Augmenture“

Bernhard Thiersch
Hochschule Reutlingen

bernhard.thiersch@student.hochschule-reutlingen.de

ABSTRACT

Am Beispiel des Spieles „Augmenture“ werden Möglichkeiten beschrieben, eine mixed-reality Schnitzeljagd für Smartphones umzusetzen. Anforderungen werden dargestellt und entsprechende Lösungswege in einem Architektur-Entwurf diskutiert. Weiterhin wird beschrieben, welche Aufgaben von AR-Browsern übernommen werden können und wie diese dazu in die Architektur integriert werden müssen.

1. EINLEITUNG

Augmenture ist ein Projekt des Studiengangs Medien- und Kommunikationsinformatik an der Hochschule Reutlingen. Der Name setzt sich aus dem englischen Begriff für erweiterte Realität, „Augmented Reality“ (AR) und Abenteuer, „Adventure“ zusammen. Projektziel ist es, eine Einführung ins Fach Informatik zu geben und dessen Attraktivität insbesondere für Schülerinnen im Alter von 15 Jahren zu erhöhen (vgl. [1]). Auf spielerische Weise sollen Inhalte und Problemlösungsmethoden aus der Informatik erkundet werden können.

2. SPIELIDEE

Die Schülerinnen absolvieren eine Schnitzeljagd. An verschiedenen Stationen müssen Rätsel gelöst werden. Wurde das Rätsel an einer Station gelöst, suchen die Schülerinnen die nächste Station. Die Rätsel enthalten Problemstellungen aus der Informatik. Um Motivation und Spielfreude zu erhöhen, konkurrieren verschiedene Gruppen beim Lösen der Aufgaben.

Eine klassische Etappe einer Schnitzeljagd besteht aus einer Station, an der Hinweise zum Finden der jeweils nächsten Station versteckt sind und dem Weg zu dieser. Im Spiel Augmenture wird den Schülerinnen an jeder Station ein Rätsel aus dem Bereich der Informatik gestellt. Die Lösung eröffnet den Weg zur nächsten Station.

Um die Informatik nicht nur in der Theorie zu erleben, sollen aktuelle Entwicklungen aus der Informatik eingesetzt werden.

3. ANFORDERUNGEN

Grundlegende Anforderungen an das aus dem gleichnamigen Projekt resultierende Abenteuerspiel Augmenture sind:

1. Der Einsatz aktueller Technologien die bei jugendlichen Schülerinnen beliebt sind.
2. Räumlich verteilte Stationen die Fragen bereithalten und Antworten annehmen. Die Interaktion soll in Echtzeit möglich sein.
3. Ein Leitsystem das die Reihenfolge der Etappen regelt. Es zeigt nur den Weg zur jeweils nächsten Station.
4. Ein zentrales Frage-Antwort-System, dass für die Zielgruppe lösbare Fragen aus dem Kontext der Informatik enthält.
5. Ein Auswertungssystem, dass mit dem Frage-Antwort-System verbunden ist und Daten über Etappen der Schnitzeljagd sammelt und evaluiert. Zusätzlich soll es jederzeit Auskunft über den aktuellen Punktestand geben.

Im Folgenden wird näher auf die Anforderungen 1, 2 und 3 eingegangen. Mit den Anforderungen 4 und 5 befassen sich Heberle und Romer in [1].

3.1 TECHNIK FÜR JUNGE LEUTE

Durch ihre hohe Attraktivität und wachsende Popularität sind Smartphones aktuelles Thema. Im vergangenen Jahr stieg der Anteil der Smartphone-Nutzer in Deutschland um 65 Prozent auf 5,1 Millionen [2]. Das sind 23 Prozent der mobilen Konsumenten. Ein Grund für die hohe Attraktivität ist die wachsende Leistungsfähigkeit der Geräte. Daraus resultieren vielseitige Einsatzmöglichkeiten. Gründe für die wachsende Popularität sind immer erschwinglicher werdende Geräte und sinkende Preise für mobiles Internet. Nach [3] sind geringe Preise weltweit wichtigstes Kaufkriterium bei Smartphones.

Nachdem Frauen die männlichen Benutzer sozialer Netzwerke in Anzahl und Aktivität übertroffen haben (vgl. [9]), zeigt sich in den Vereinigten Staaten eine ähnliche Entwicklung auch bei der Benutzung von Smartphones: von den Smartphone-Besitzern im Alter zwischen 15 und 24 Jahren sind 55 Prozent weiblich. In Deutschland waren es im Dezember 2010 34 Prozent in dieser Altersgruppe [3]. Bei einer Befragung der Aris im März 2011 gaben 20 Prozent der befragten Frauen an, dass sie

das mobile Internet mit Handys nutzen, während es bei den Männern nur 16 Prozent waren [4].

Hieraus wird deutlich, dass sich der Einsatz von Smartphones eignet um Anforderung 1 (aktuelle, beliebte Technologie) zu treffen. Für Augmentare stehen Smartphones vom Typ „HTC Desire“ und „HTC Desire Z“ mit der Plattform Android zur Verfügung.

3.2 INTERAKTIVE STATIONEN

Die Schnitzeljagd Augmentare soll die Schülerinnen am Eingang des Campus abholen und über verschiedene Orte des Campus schließlich zum Informatik-Gebäude führen. Im Gebäude werden weitere Rätsel gestellt. So gibt es sowohl Stationen im Freien als auch innerhalb von Gebäuden.

Stationen können in Augmentare aus stationären Computern bestehen, an denen die Schülerinnen Rätsel lösen. Solche Stationen haben hohen Hardwarebedarf, sind aufwändig in der Wartung und schwierig außerhalb von Gebäuden zu verwenden. Die Nutzung von Smartphones ermöglicht es, Stationen durch Verwendung ortsbezogener Dienste (siehe 5) zu realisieren. Solche Stationen sind zentral wartbar und witterungsunabhängig. Auch die Auswertung der Antworten (Anforderung 5) kann so ohne zusätzlichen Kommunikationsaufwand an zentraler Stelle erfolgen.

3.3 INTERAKTIVE WEGFÜHRUNG

Das Leitsystem (Anforderung 3) benötigt Daten über den aktuellen Fortschritt des Spielers, da es nur den Weg zur nächsten Station offenbaren soll. Abgesehen davon hat es bei Augmentare dieselbe Funktion wie in klassischen Schnitzeljagden. Es gibt den Spielern Orientierungshilfe beim Finden der nächsten Station. Für Augmentare eignet sich ein System, das Kenntnis über die aktuelle Position des Gerätes hat und interaktiv Orientierungshilfen für Fußgänger zur Verfügung stellt. Frühe Ansätze solcher Systeme sind in [5] und [6] dargestellt und in [7] um verschiedene Ausgabearten erweitert und ausgewertet. Den Einsatz von Augmented-Reality als Orientierungshilfe für Fußgänger beschrieb [8]. Häufig sind Fußgängerleitsysteme als Zusatz im Lieferumfang von Fahrzeug-Navigationssystemen enthalten. Inzwischen gibt es einige Anbieter von Fußgängernavigationssystemen für Smartphones.

Orientierungshilfen in Augmentare können neben Kartennavigation auch zum Beispiel virtuelle Pfeile auf dem Kamerabild sein, die in die Richtung der nächsten Station zeigen und eine Entfernungsangabe, die die Distanz zur nächsten Station in Metern angibt.

4. POSITIONSBESTIMMUNG

Zur Nutzung ortsbezogener Dienste ist es nötig, die eigene Position bestimmen zu können. Durch Messung der Laufzeit von Funksignalen, können Rückschlüsse auf die Entfernung von Funkwellensendern gezogen werden. Wurde die Entfernung zu mehreren Sendern ermittelt deren geographische Position bekannt ist, kann die eigene Position mit Hyperbelberechnungen bestimmt werden. Eingesetzt werden solche Multilaterationsverfahren zum Beispiel bei Satellitennavigationssystemen (GPS, GLONASS, Galileo), WPS (wireless positioning systems) GSM-Ortung über Funkzellen und RFID-Ortung. Die aktuellen Smartphones der Firmen Apple und HTC verfügen über GPS-Empfänger und WLAN-Funkmodems. Die Betriebssysteme iOS und Android bieten Anwendungsschnittstellen für die Positionsbestimmung mit GPS und WPS. Für Augmentare sind demnach alle drei Positionsbestimmungsarten nutzbar.

5. ORTSBEZOGENE DIENSTE

Ein ortsbezogener Dienst (englisch: location based service, LBS), ist ein Dienst der einem Nutzer auf Anfrage in Kenntnis und in Abhängigkeit seines geographischen Standorts erbracht wird. Dienstanutzer sind bei Augmentare die Spieler. Der ortsabhängige Dienst liefert ihnen die für sie aktuelle Station.

5.1 LBS-INTERAKTIONSSCHEMA BEI SMARTPHONES

Das grundlegende Interaktionsschema für ortsbezogene Dienste wird in [10] beschrieben und im Folgenden für die Nutzung mit Smartphones angepasst.

Dienstanfrage:

Benutzer fordern den jeweiligen Dienst über eine Client-Anwendung auf dem Smartphone an. Die Client-Anwendung ermittelt die aktuelle Position des Gerätes und sendet diese zusammen mit der Dienstanfrage über ein Netzwerk (UMTS, GSM, WLAN, Internet) an die Server-Anwendung des Diensteanbieters.

Diensterbringung:

Die Server-Anwendung des Diensteanbieters verarbeitet die Positionsangabe zur Auswahl der geforderten Informationen und antwortet dem Dienstanutzer über das Netzwerk, indem sie den Dienst zur Verfügung stellt.

5.2 CLIENT-ANWENDUNG

Die Client-Anwendung von Augmentare soll die jeweils aktuelle Station bei der Server-Anwendung erfragen können. Zudem soll sie die Interaktion mit

der ermittelten Station ermöglichen (vgl. 3 Anforderung 2).

Hierzu muss sie die geographische Position des Smartphones abfragen können und diese dann an die Server-Anwendung weiterleiten. Als Antwort erhält sie die Daten der entsprechenden Station mit dem dazugehörigen Rätsel. Zur Bearbeitung des Rätsels müssen sowohl eine geeignete Ausgabe der Frage als auch die Eingabe von Antworten möglich sein. Die Antworten werden an die Server-Anwendung gesendet.

Idealerweise übernimmt die Client-Anwendung auch Aufgaben der Wegführung, indem sie Unterstützung beim Finden der nächsten Station bietet (vgl. 3 Anforderung 3).

5.3 SERVER-ANWENDUNG

Aufgabe der Server-Anwendung bei Augmenture ist die Bereitstellung von Stationen in Abhängigkeit der geographischen Position der Smartphones und der erreichten Etappe. Zudem Repräsentation der Kernfunktionalität des Frage-Antwort-Systems sowie des daran gekoppelten Auswertungssystems. Dazu gehört, dass die Server-Anwendung die Daten für Stationen, Fragen und Spielerstände verwaltet.

5.4 AR-BROWSER

Aktuelle Werkzeuge für die Umsetzung von ortsbezogenen Diensten sind sogenannte „Augmented Reality Browser“ (AR-Browser). AR-Browser-Systeme sind dienstorientierte Architekturen für Anbieter und Benutzer von ortsbezogenen Inhalten. Neben der Dienstvermittlung stellen sie die Client-Software für gängige Smartphone-Plattformen und bieten Programmierschnittstellen für Anbieter von ortsbezogenen Inhalten. Ortsbezogene Inhalte sind

als sogenannte „Points of Interest“ (POI) strukturiert. POIs sind georeferenzierte Informationen. Benutzer können diese in Kartenansicht, Listenansicht oder in erweiterter Realität direkt in das aktuelle Kamerabild eingebunden betrachten. Damit diese Ansichten möglich sind, muss das Smartphone über einen GPS-Empfänger, Kompass-Sensoren und eine Kamera verfügen. Eine ausführliche Beschreibung und Gegenüberstellung aktueller AR-Browser ist in [11] gegeben.

5.4.1 VORTEILE FÜR AUGMENTURE

AR-Browser haben einige Eigenschaften, die sie für die Verwendung in Augmenture eignen. Sie stellen eine Architektur für ortsbezogene Dienste bereit. Sie liefern eine Client-Anwendung und skizzieren durch Schnittstellen die Kommunikation mit der Server-Anwendung. AR-Browser delegieren die Positionsermittlung an Betriebssystem-Schnittstellen und formulieren daraus Anfragen an die Server-Anwendung. Sie definieren weitere Eigenschaften von geographischen Punkten und bieten Navigationshilfen um diese zu finden. Anforderung 2 (interaktive Wegführung), kann dadurch erfüllt werden.

Somit kann ein AR-Browser einige Aufgaben unserer Client-Anwendung (vgl. 5.2) übernehmen. Die AR-Browser Layar¹², Junaio¹³ und Wikitude¹⁴ sind zudem sowohl für Apple iOS als auch für Android verfügbar. Durch ihre Verwendung wäre Augmenture plattformübergreifend nutzbar.

¹² <http://www.layar.com/>

¹³ <http://www.junaio.com/>

¹⁴ <http://www.wikitude.org/>

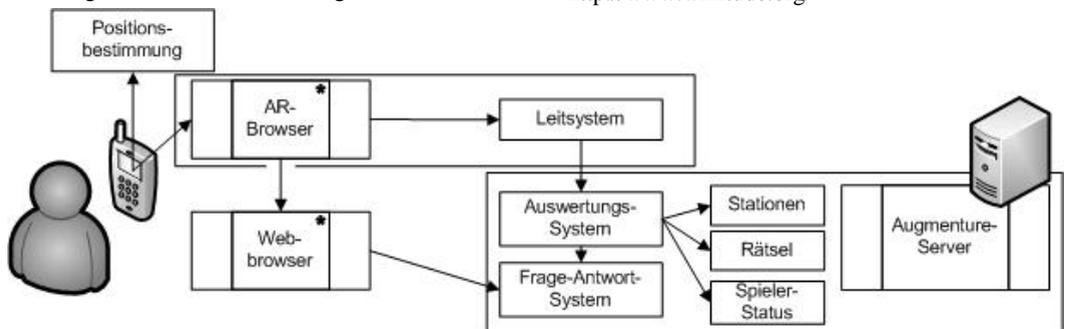


Abbildung 1: Architektur Augmenture

5.4.2 NACHTEILE FÜR AUGMENTURE

Die Präsentation von Rätselfragen, in einer für Augmenture sinnvollen Weise und die Annahme der Antworten, können AR-Browser nicht übernehmen. Ein POI kann nur als Bild oder - je nach AR-Browser - 3D-Objekt präsentiert werden. Zudem sind die Interaktionsmöglichkeiten mit POIs auf „ausgewählt“ und „nicht ausgewählt“ beschränkt. Es wäre denkbar, Fragen durch POI-Grafiken zu repräsentieren und Antwortmöglichkeiten als zusätzliche POIs zu gestalten, von denen der richtige ausgewählt werden muss. Doch durch Ungenauigkeiten bei Positionsbestimmung und Registrierung (Verankerung des virtuellen Objekts vor dem realen Hintergrund) ist es sehr schwierig, mehrere POIs gleichzeitig, aus der richtigen Perspektive und in ausreichender Dauer anzuzeigen. Da dieser Ansatz dadurch Gefahrenpotenzial für den Spielspaß birgt und zudem die Freiheit bei der Rätselstellung stark einschränkt, wurde er verworfen.

5.4.3 KOMBINATION MIT WEBPAGES

Um weitere Informationen zu einem POI bekommen zu können, erlauben AR-Browser das Setzen von Weblinks. Der Klick auf den Weblink eines POIs öffnet ein Fenster des Webbrowsers des Smartphones. Schließt man dieses, ist wieder die Oberfläche des AR-Browsers im Vordergrund.

Augmentures Frage-Antwort-System kann dadurch vom AR-Browser entkoppelt und über Webpages realisiert werden. Webbrowser bieten die für die Rätsel nötigen Interaktionsmöglichkeiten.

Antworten können direkt vom Auswertungssystem abgelesen und bearbeitet werden. Die lose Kopplung von Frage-Antwort-System und Stationen vereinfacht zudem die Wartbarkeit von Augmenture. Abbildung 1 zeigt die resultierende Architektur.

5.5 AUSSENBEREICH

Für Augmenture werden die AR-Browser Layar und Junaio verwendet. Vergleiche mit Test-POIs und WLAN-Verbindung ergaben, dass Layar auf dem Smartphone HTC Desire und der Plattform Android genauere Resultate bei der Registrierung der virtuellen Objekte auf dem Bildschirm erzielt als Junaio. Aus diesem Grund wird für die Augmenture-Etappen die sich im Freien befinden, Layar benutzt.

Layar nutzt zwar laut Herstellerangaben zusätzlich zu GPS auch WPS (Trilateration mit kartographierten WLAN-Access-Points), doch die Ergebnisse innerhalb des Informatik-Gebäudes sind für Augmenture unbrauchbar. Faktoren für die Genauigkeit von WPS sind Hindernisse wie Wände, Aktualität

des Kartenmaterials und die eingesetzten Berechnungsverfahren.

Eine Gegenüberstellung der Programmierschnittstellen von Layar und Junaio findet sich in [7].

5.6 INNENBEREICH

Nach Herstellerangaben verzichtet Junaio auf WPS, doch es bietet innerhalb von Gebäuden eine andere Möglichkeit: optisches Tracking. Layar bietet diese Möglichkeit nicht. Laut Hersteller befindet sich diese in der Entwicklung.

Durch sogenannte LLA-Marker (Latitude, Longitude, Altitude), können geodätische Daten kodiert werden. LLA-Marker können im Gebäude angebracht werden und werden von Junaio optisch ausgewertet. Die im Marker enthaltene geographische Positionsangabe wird dann von Junaio als aktueller Standort interpretiert. Eine detaillierte Beschreibung der Tracking-Methoden von Junaio liefert [13]. Zusätzlich zur optischen Repräsentation von geographischen Koordinaten können mit Junaio auch Mediendaten (3D-Objekte, Audio, Video) grafisch referenziert werden. Auf diese Methode geht [13] ein.

6. FAZIT

Es wurde gezeigt, dass die Umsetzung einer mixed-reality Schnitzeljagd im Freien und in Gebäuden mit aktuellen Werkzeugen möglich ist. Genauer wurde gezeigt, wie Anforderung 1 (Einsatz aktueller Technologien), durch die Nutzung von Smartphones erfüllt werden kann. Anforderung 2 (räumlich verteilte Stationen), kann durch die Kombination von AR-Browser und Webbrowser realisiert werden. Durch lose Kopplung ergeben sich dazu Vorteile bei der Wartbarkeit.

Anforderung 3 (Führungs-System), braucht Unterstützung durch die Server-Anwendung, da der aktuelle Fortschritt des Spielers von ihr verwaltet wird. Diese sorgt dafür, dass nur die jeweils nächste Station als POI erscheint. Durch integrierte Navigationshilfen unterstützen AR-Browser aber bei der Wegfindung. Pfeile, die jedoch die Geländebeschaffenheit und mögliche Hindernisse mit einbeziehen, müssten als eigene zusätzliche POIs hinterlegt werden.

Eine pädagogisch-didaktische Behandlung der Anforderungen 4 (Frage-Antwort-System) und 5 (Auswertungssystem), liefern [1]. Die technischen Grundlagen für die Umsetzung beider Systeme sind durch die Architektur mit Webpages und Datenverwaltung gegeben.

Es ist abzusehen, dass die Vielfalt von AR-Browsern wachsen wird. Zur Verwaltung von AR-Inhalten gibt es verschiedene Entwicklungen [11],

deren Funktionsumfang für Augmenture jedoch noch nicht ausreicht (z.B. fehlen Spielerverwaltung und Auswertungssystem). Mit Argon¹ kam im Februar 2011 ein AR-Browser-System auf den iPhone-Markt, das mobile Augmented-Reality mit KML, HTML, JavaScript und CSS standardisiert. Wenn die Portierung auf Android abgeschlossen ist, sollte die Eignung von Argon für Augmenture geprüft werden.

Der erste Durchlauf von Augmenture mit Schülerinnen ist im Sommer dieses Jahres geplant.

Anschließend wird das Spiel evaluiert. Die Ergebnisse fließen in einen Verbesserungsprozess ein. Dies betrifft das Software- und das Spielkonzept. Ob Augmenture die erwünschte langfristige Wirkung, die Attraktivität der Informatik bei Schülerinnen zu erhöhen, erzielt, muss über längeren Zeitraum beobachtet werden. Hierzu bieten sich Befragungen nach einer Zeit von etwa zwei Jahren an.

7. LITERATURVERZEICHNIS

- [1] Heberle, C., Romer, B. Didaktische- und Spielkonzeptionelle Aspekte einer Augmented-Reality-Schnitzeljagd am Beispiel „Augmenture“. Tagungsband zur Informatics Inside 2011: Grenzen überwinden -Virtualität erweitert Realität. Hochschule Reutlingen.
- [2] ComScore Inc., MobiLens. Januar 2011. Smartphones gewinnen an Fahrtwind in Deutschland, Marktstudie. http://www.comscore.com/ger/Press_Events/Press_Releases/2011/1/Google_Android_Shows_Fastest_Growth_Among_Smartphone_Platforms_in_Germany.
- [3] Nielsen. Dezember 2010. Mobile Youth Around The World, Marktstudie. <http://www.nielsen.com/us/en/insights/reports-downloads/2010/mobile-youth-around-the-world.html>.
- [4] BITKOM. März 2011. Zahl der Handy-Surfer in einem Jahr verdoppelt. http://www.bitkom.org/de/presse/8477_67381.aspx.
- [5] Ertan S, Lee C, Wollets A, Tan H, Pentland A. 1998. A wearable haptic navigation guidance system. Second international symposium on wearable computers. Pittsburgh.
- [6] Thomas B, Demczuk V, Piekarski W, Hepworth D, Gunther, B. 1998. A wearable computers system with augmented reality to support terrestrial navigation. Second international symposium on wearable computers. Pittsburgh.
- [7] Stark A., Riebeck M., Kawalek J. 2007. How to Design an Advanced Pedestrian Navigation System: Field Trial Results. IEEE Int. Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, Dortmund, Germany
- [8] Feiner S, MacIntyre B, Hollerer T, Wester A. 1997. A touring machine: prototyping 3D mobile augmented reality systems for exploring the urban environment. Proceedings of the first international symposium on wearable computers
- [9] Lee, E. März 2011. Why Women Rule The Internet. <http://techcrunch.com/2011/03/20/why-women-rule-the-internet/>.
- [10] Spreitzer, M.; Theimer, M. 1993. Providing location information in a ubiquitous computing environment. ACM SOSP '93 Proceedings of the fourteenth ACM symposium on Operating systems principles, 1993
- [11] Hauser, D. 2011. Konzeption und prototypische Umsetzung einer Rich Internet Application zur Vereinfachung der Integration von Inhalten in einen mobilen Augmented Reality Browser, Master-Thesis, Hochschule Reutlingen
- [12] Kallenberg, P. 2001. Programmierschnittstellen von Augmented-Reality-Browsern im Rahmen des Projekts „Augmenture“. Tagungsband zur Informatics Inside 2011: Grenzen überwinden -Virtualität erweitert Realität. Hochschule Reutlingen.
- [13] Schultheiß, R. 2011. Mobile Augmented Reality auf Smartphones mit dem AR-Browser Junaio. Tagungsband zur Informatics Inside 2011: Grenzen überwinden -Virtualität erweitert Realität. Hochschule Reutlingen.

¹ <http://argon.gatech.edu/index.html>

Didaktische und spielkonzeptionelle Aspekte einer Augmented-Reality-Schnitzeljagd am Beispiel „Augmenture“

Benjamin Romer
Hochschule Reutlingen

benjamin.romer@student.reutlingen-
university.de

Christine Heberle
Hochschule Reutlingen

christine.heberle@student.reutlingen-
university.de

ABSTRACT

Um Mädchen für das Thema Informatik zu interessieren, wurde an der Hochschule Reutlingen das „Augmenture“-Projekt gestartet. Durch eine virtuelle Schnitzeljagd auf Smartphones soll einer Gruppe von jugendlichen Mädchen im Alter von ca. 15 Jahren (Klasse 9) das Thema Informatik auf spielerische Weise nahe gebracht werden. Diese Ausarbeitung beschreibt die Zielgruppenanalyse und das Vorgehen bei der Entwicklung möglicher Aufgaben für die Schnitzeljagd. Ebenso wird auf das Punkte- und Bewertungssystem eingegangen. Es wurde ein Kontakt zum Isolde-Kurz Gymnasium in Reutlingen aufgebaut. Mit Schülerinnen dieser Schule ist Ende Juli 2011 ein Durchlauf geplant.

1. EINLEITUNG

MINT bedeutet Mathematik, Informatik, Naturwissenschaften und Technik. Der Begriff MINT-Berufe bezeichnet somit die Berufsgruppen, die sich mit diesen vier Bereichen befassen. Der Frauenanteil in diesen Berufen ist auch heute noch geringer als der Männeranteil. So lag der Prozentanteil an Absolventinnen im Jahr 2009 im Technikbereich bei 7,1% und in den MINT-Bereichen bei 12,8 %. Mit Initiativen im MINT-Bereich soll bis 2015 eine Frauenquote von 32 % bei den Erstabsolventen in den MINT-Bereichen erreicht werden. [24]

Um den Frauenanteil zu steigern, gibt es MINT-Projekte, mit denen versucht wird, in jungen Frauen das Interesse an Mathematik, Informatik, Naturwissenschaften oder Technik zu wecken. Diese MINT-Projekte sind unter anderem auf der Website des nationalen Paktes für Frauen in MINT-Berufen aufgeführt. [25]

Diese Ausarbeitung untersucht die Gründe für den geringen Frauenanteil an den MINT-Berufen und stellt das Projekt „Augmenture“ mit Konzeption und Umsetzung vor. Mit dem Projekt wird versucht, das Thema Informatik für Mädchen attraktiver zu machen.

Der Bericht gliedert sich in folgende Bereiche:

- Einführung in das Thema
- Einordnung des geplanten Spiels „Augmenture“ in bildungstechnische Konzepte
- Konzeption mit Zielgruppenanalyse und Ausarbeitung von Route und Aufgaben
- Geplante Evaluation mit Schülergruppen
- Fazit und Ausblick

1.1 PROBLEMSTELLUNG

Aufgabe des Teams um Augmenture ist es, ein Spiel zu entwickeln, das Schülerinnen einen Einblick in die Informatik vermittelt. Hierbei ist der Bezug zur Informatik nicht offen ersichtlich, sondern wird nach Beendigung des Spieles in einer Nachbesprechung erarbeitet.

Die Motivation hinter Augmenture ist, das Interesse von jungen Frauen an informatisch ausgerichteten Themen zu wecken und sie dazu zu motivieren, sich damit auseinander zu setzen. Scheu und falsche Vorstellungen vom Thema Informatik sollen genommen werden.

1.2 PROJEKTBESCHREIBUNG

1.2.1 ANDERE PROJEKTE

Es gibt eine Vielzahl von Projekten die sich damit befassen, Mädchen für MINT-Themen zu begeistern. Beispiele hierfür sind:

- taste MINT
- Fraunhofer Talent School und Junior-Ingenieur-Akademie
- MINT Role Models

Laut [25] gibt es in Deutschland über 800 Angebote und Projekte zu diesem Thema.

1.2.2 AUGMENTURE

Augmenture ist eine Augmented-Reality-Schnitzeljagd, die auf dem Gelände der Hochschule Reutlingen stattfindet. Stationen sind auf dem Campus und im Informatikgebäude verteilt. An jeder Station der Schnitzeljagd bekommen die

Teilnehmer eine Aufgabe oder eine Information zum Thema Informatik. Ziel des Spieles ist es, die Aufgaben richtig zu lösen und so viele Punkte wie möglich zu sammeln. Das Logo von Augmenture ist in Abbildung 1 zu sehen.



Abbildung 1: Augmenture-Logo

Das Projekt ist zum Zeitpunkt dieses Beitrags noch nicht abgeschlossen. Die Fertigstellung ist bis Ende April 2011 geplant. Der Testlauf mit einer Schülergruppe des Isolde-Kurz-Gymnasiums in Reutlingen ist an einem Tag des Zeitraums vom 18. – 22. Juli 2011 geplant.

1.2.3 GRUNDLAGEN

Augmenture wird mit Hilfe eines Smartphones mit dem Betriebssystem Android gespielt. Zum Spielen ist eine Augmented-Reality¹-Anwendung, die Junaio-App, der Firma Metaio² notwendig. Weitergehende Informationen zu den Programmierschnittstellen und zu mobiler AR sind in den Artikeln von Kallenberg in [26] und Schulheiß in [27] nachzulesen. Entwurfsentscheidungen der Anwendung werden von Thiersch in [28] diskutiert.

2. EINORDNUNG AUGMENTURE

Augmenture ist ein Spiel, das den Spielenden unbewusst einen Einblick in die Informatik bieten soll. Aus diesem Grund gehört Augmenture zum Bereich des Game-Based Learnings³ - genauer: zum Digital Game-Based Learning (kurz: DGBL), da es sich um ein Spiel handelt, das mit Hilfe von digitalen Medien (in diesem Fall ein Smartphone mit dem Betriebssystem Android) gespielt wird. Augmenture gehört zum Bereich des E-Learnings. Das Spiel vermittelt mit Hilfe eines elektronischen Mediums Wissensinhalte. Abbildung 2 zeigt die Beziehungen zwischen Serious Games und ähnlichen Bildungskonzepten nach Breuer und Bente [29].

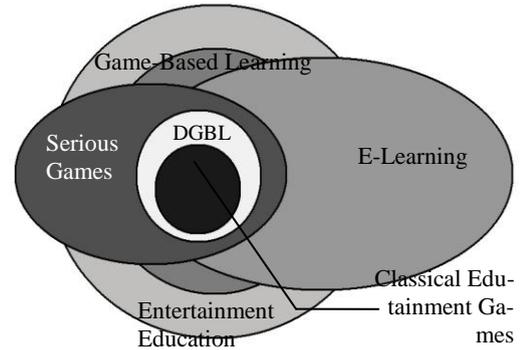


Abbildung 2: Beziehung zwischen Serious Games und ähnlichen Bildungskonzepten nach Breuer und Bente [29]

Ausgehend von dieser Abbildung lässt sich sagen, dass Augmenture zum Bereich „Entertainment Education“⁴ gehört.

3. KONZEPTION

Im Folgenden wird die Konzeptionsphase von Augmenture beschrieben mit Zielgruppenanalyse, Aufgabenentwicklung, Belohnungs- und Punktesystem und Avatar.

3.1 ZIELGRUPPENANALYSE

3.1.1 MÄDCHEN UND INFORMATIK

Im Rahmen der Zielgruppenanalyse wurde eine Recherche zum Thema Mädchen und Informatik durchgeführt. Hierbei wurde betrachtet, was Mädchen über Informatik denken, welche Vorstellungen sie davon haben und welche Vorurteile es gibt. Das Ergebnis der Recherche wird im Folgenden dargestellt.

Laut García-Crespo et al. wird die IT als harte, „uncoole“ Arbeit angesehen. Vorurteile sind unter anderem, dass Informatiker optisch unattraktiv sind, sich schlecht kleiden, in Kellerzimmern arbeiten und sozial inaktiv sind. [30]

Eine kurze Befragung von Mädchen der Klassenstufe 9-10 ergab, dass diese Vorurteile („wie die schon aussehen!“), auch heute noch bestehen.

Beyer et al. [31] nennen zwei Gründe dafür, dass Informatik für Frauen nicht attraktiv ist. Ein Grund sind negative Klischees über die Informatik und die Menschen, die damit arbeiten. So werden Informatiker zwar als „intelligent“ bezeichnet, aber ihnen wird auch ein Mangel an sozialen Kompetenzen zugeschrieben.

Der andere Grund ist das niedrige Selbstvertrauen von Frauen im Bezug auf Informatikthemen.

¹ Kurz: AR; dt. „erweiterte Realität“

² <http://www.metaio.de/>

³ Dt. „spielbasiertes Lernen“

⁴ Dt. etwa „Unterhaltungsbildung“

Frauen denken nach Beyer et al., dass sie in männlichen Domänen, wie z.B. Chemie, Mathematik und Informatik nicht so erfolgreich sein werden, wie ihre männlichen Mitstreiter.

Werden Frauen im Bereich Informatik von der Familie und vom Freundeskreis gefördert, so ist es wahrscheinlicher, dass sie einen Abschluss in Informatik bekommen. [31]

3.1.2 FRAGEBOGEN UND AUSWERTUNG

Unsere Zielgruppe sind Mädchen der Jahrgangsstufe 9. Um möglichst viele Informationen über diese Gruppe zu erhalten, wurde ein Fragebogen erstellt. Tabelle 1 zeigt einen Ausschnitt aus diesem Fragebogen mit Antworten. Mit den Erkenntnissen ist ein Zielgruppen-Profil erstellt worden.

Tabelle 1: Ausschnitt aus Fragebogen mit Informationen zu der Zielgruppe

Soziodemographische Merkmale der Zielgruppe	
Alter:	14 bis 16 Jahre
Geschlecht:	Weiblich
Taschengeld:	10 bis 20 Euro
Tätigkeit:	Schülerinnen
Schule:	Gymnasium
Psychologische Merkmale der Zielgruppe	
Lernen:	Belohnungen fördern den Lerneffekt.
Schule:	ca. 50 % der befragten Schülerinnen haben Probleme mit den Fächern Mathematik und Physik.
Denken übers Studieren:	Nur schlaue Leute. Brillenträger. Studieren ist schwierig.
Soziologische Merkmale der Zielgruppe	
Gruppe:	Freundeskreis besteht im Durchschnitt aus 6 Personen.
Nebengruppen:	Freunde aus dem Sportverein. Treffen 1- bis 2-mal pro Woche. Austausch von Informationen (Jungs, Gerüchte, Lästern).
Rollenverteilung:	1 bis 2 beste Freundinnen. Geheimnisse werden anvertraut.

Der Fragebogen enthält noch weitere soziologische Merkmale:

- **Soziale Netzwerke:** Facebook, Kwick, SchülerVZ
- **Sportinteressen:** Tennis, Volleyball, Turnen, Ski, Snowboard
- **Freizeitbeschäftigungen:** Freunde treffen, PC (Soziale Netzwerke), shoppen
- **Lieblingsbücher/Genre:** Romane
- **Lieblingsfilme/Genre:** Liebesfilme, Komödien
- **Lieblingsmusik/Genre:** Pop, Charts
- **Lieblingsschulfach:** Sport, Mathematik, Mensch und Umwelt
- **Zukunft:** noch keine konkreten Vorstellungen
- **Berufswahl:** Mediendesign, noch unentschieden

3.2 ROUTE

3.2.1 AUßEN

Die Route auf dem Campus besteht aus elf Punkten. Die Spieler werden unter anderem zu einem Studentenwohnheim, zur Mensa, zur Bibliothek und zur Aula geführt. An acht der insgesamt elf Punkte werden Aufgaben gestellt. Die Aufgaben sind abhängig von der GPS-Position des Spielers und werden je nach Position aufgerufen.

3.2.2 INNEN

Im Gegensatz zu der Route auf dem Campus, werden im Informatikgebäude weniger Aufgaben gestellt. Die Route besteht aus zehn Punkten, die von den konkurrierenden Gruppen in unterschiedlicher Reihenfolge abgelaufen werden. An fünf dieser Punkte werden die Spieler mit Aufgaben konfrontiert. Die Aufgaben werden mit Hilfe von im Gebäude verteilten Markern aufgerufen. Diese Marker werden vom Smartphone erkannt und lösen eine Aktion aus. In diesem Fall wäre die Aktion das Aufrufen der entsprechenden Aufgabe.

3.3 AUFGABEN

3.3.1 RECHERCHE AUFGABEN

Für die einzelnen Punkte wurden Aufgaben recherchiert, die von den Schülergruppen zu lösen sind. Die Aufgaben mussten folgende Kriterien erfüllen:

- Die Aufgaben sollen einen Bezug zur Informatik haben

- Die Aufgaben sollen von der Zielgruppe gelöst werden können
- Der Schwierigkeitsgrad soll angemessen sein und die Aufgaben sollen in ungefähr 5 – 10 Minuten gelöst werden können.
- Die Aufgaben sollen weitgehend ohne weitere Hilfsmittel lösbar sein.

Als schon vorhandene Aufgabensätze wurden die Aufgaben des Informatik-Bibers und von Computer Science Unplugged gefunden (siehe Kapitel 3.3.2 und 3.3.3) [32][33].

3.3.2 INFORMATIK-BIBER

Seit 2006 gibt es den deutschen Informatik-Wettbewerb „Informatik-Biber“. Er ist durch eine Partnerschaft mit der internationalen Wettbewerbs-Initiative „Bebras“ (dt.: Biber) entstanden. Initiiert wurde das Projekt „Bebras“ durch die litauische Didaktikerin Valentina Dagiéne. [34]

Der Informatik-Biber soll Jugendliche früh auf Informatik und ihre vielfältigen Aspekte aufmerksam machen und für die Beteiligung an Informatik-Wettbewerben gewinnen. Wie alle Bebras-Wettbewerbe wird er als Quiz durchgeführt. Der Informatik-Biber konnte schnell hohe Teilnahmezahlen erreichen.

Die Teilnahmezahlen von 2009 und 2010 nach Jungen und Mädchen getrennt zeigen allerdings, dass der Mädchenanteil nach der 10. Klassenstufe im Jahr 2009 von 41,4% auf 30,7% und im Jahr 2010 von 40,0% auf 29,1% gesunken ist. [33]

Zielsetzung

Die Veranstalter verfolgen mit der Durchführung des Informatik-Bibers folgende Ziele [34]:

- Begegnung mit Informatik
- Werbung für Informatik
- Frühe Ansprache
- Heranführung an Informatik-Wettbewerbe
- Informatik-Begabung identifizieren

Mit diesen Zielen soll den Jugendlichen in erster Linie die Vielfalt der Informatik altersgerecht präsentiert werden. Speziell durch die frühe Ansprache sollen die Jugendlichen frühzeitig auf Informatik aufmerksam gemacht werden. Dadurch soll rechtzeitig den Klischees und Vorurteilen bezüglich der Informatik entgegen gewirkt werden [34]. Eine besondere Herausforderung stellen hierbei die Mädchen dar, die schon sehr früh diese Vorurteile übernehmen. Einige dieser Vorurteile wurden in Kapitel 3.1.1 genannt.

Die Aufgaben des Informatik-Biber-Wettbewerbs zeichnen sich durch folgende Eigenschaften aus:

- **Keine Vorkenntnisse:** Da der Informatik-Biber eine Großzahl an Teilnehmern erreichen soll, dürfen die Aufgaben keine Vorkenntnisse erfordern.
- **Struktur der Aufgaben:** Die Aufgaben bestehen aus einer Erläuterung, der Fragestellung und Antwortmöglichkeiten. In der Erläuterung sind Angaben, die zum Lösen der Aufgabe erforderlich sind. Die Fragestellung bezieht sich auf die Erläuterung. Als Antwortmöglichkeiten gibt es meist eine Auswahl von vier Antworten.
- **Fragestellung und Antwortform:** Die Fragestellung muss eindeutig sein, damit die Aufgabe schnell erfasst werden kann. In der Beispielaufgabe sind vier Antwortmöglichkeiten geboten. Genau eine davon ist richtig. Hierbei handelt es sich um Multiple-Choice Aufgaben, die beim Informatik-Biber bevorzugt werden. Die Wettbewerbsteilnehmer des Informatik-Bibers haben nicht viel Zeit um eine Aufgabe zu lösen. Ungefähr zwei Minuten wird für eine Aufgabe berechnet. Deshalb müssen die Aufgaben kurz sein.

Beispielaufgabe

Im nächsten Abschnitt wird die Beispielaufgabe „Freunde im Netz“ aus dem Informatik-Biber 2009 vorgestellt. Die Antwortmöglichkeiten sind in Abbildung zu sehen.

- Michael ist mit Leonie, Jonas und Patrick befreundet.
- Jonas ist mit Michael und Anne befreundet.
- Anne ist mit Jonas befreundet.
- Patrick ist mit Michael und Leonie befreundet.
- Leonie ist mit Michael und Patrick befreundet.

Für jede Person wird ein Punkt gezeichnet. Sind zwei Personen befreundet, werden ihre Punkte durch eine Linie verbunden.

Welches Netz ergibt sich für Michael, Leonie, Jonas, Patrick und Anne?

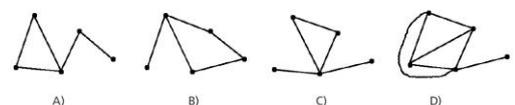


Abbildung 3: Beispielaufgabe “Freunde im Netz” aus dem Informatik-Biber 2009 [33]

3.3.3 CS UNPLUGGED

CS Unplugged steht für Computer Science Unplugged und bezeichnet Aufgaben und Übungen zum Thema Informatik, die unabhängig vom Rechner gelöst werden. [32]

Von der Verwendung der CS Unplugged Aufgaben wurde weitgehend abgesehen. Die Aufgaben erfordern meist zusätzliche Hilfsmittel und sind zeitaufwändig. Lediglich die Aufgabe zum Ausrechnen der Prüfziffer einer ISBN wurde in abgeänderter Version verwendet.

3.3.4 EIGENE AUFGABEN

Für das Projekt „Augmenture“ wurden hauptsächlich Aufgaben aus dem Informatik-Biber verwendet. Diese wurden an die Route der „Schnitzeljagd“ angepasst und verändert. Nehmen wir die Beispielaufgabe aus dem Kapitel 3.3.2. Diese Aufgabe wurde so abgeändert, dass sie zu unserem Campus passt. Der nachfolgende Text und Abbildung 4 zeigen die abgeänderte Version der Informatik-Biber Aufgabe mit möglichen Lösungen.

Der Campus ist vernetzt

- Das Rechenzentrum ist mit Gebäude 9, Gebäude 3 und Gebäude 5 verbunden.
- Gebäude 3 ist mit dem Rechenzentrum und Gebäude 4 verbunden.
- Gebäude 4 ist mit Gebäude 5 verbunden
- Gebäude 5 ist mit dem Rechenzentrum und Gebäude 9 verbunden
- Gebäude 9 ist mit dem Rechenzentrum und Gebäude 5 verbunden.

Für jedes Gebäude wird ein Punkt gezeichnet.

Stehen zwei Gebäude in Beziehung, werden ihre Punkte durch eine Linie verbunden.

Welches Netz ergibt sich für das Rechenzentrum, Gebäude 9, Gebäude 3, Gebäude 5 und Gebäude 4?

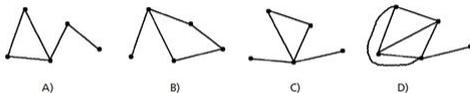


Abbildung 4: Veränderte Beispielaufgabe des Informatik-Bibers für das Projekt „Augmenture“

Hierbei wurden die Namen der Freunde durch die Gebäudenummern des Campus ausgetauscht. Die Lösungsmöglichkeiten bleiben in diesem Fall gleich. Alle weiteren Aufgaben wurden in ähnlicher Weise ortsabhängig verändert. So ist beispielsweise eine Aufgabe neben einem großen Baum auf dem Campus eine Baumstrukturaufgabe. Da es zwei Spielergruppen gibt, die zeitgleich auf der gleichen Route starten, gibt es zwei vergleichbare

Aufgabensätze. Diese unterscheiden sich zum Teil nur durch das Vertauschen der Antwortmöglichkeiten.

Desweiteren wurden zwei Programmieraufgaben konzipiert, um den Schülerinnen erste Einblicke in die Programmierung zu vermitteln. Hierfür wurden spezielle Programmiersprachen für Kinder und Jugendliche untersucht. Solche Programmiersprachen sind LOGO und Scratch.

LOGO

LOGO ist eine Programmiersprache, mit der es möglich ist, durch Befehle und Anweisungen eine grafische Schildkröte zu steuern. [35]

Ein einfacher Befehl lautet:

- vorwärts 100

Das Wort „vorwärts“ ist hier das Befehlswort und der Wert 100 der Parameter. In diesem Fall würde sich die Schildkröte um 100 Einheiten nach vorne bewegen.

Ein weiterer Befehl ist:

- rechts 90

Dieser lässt die Schildkröte auf der Stelle 90° nach rechts drehen.

Unsere Aufgabe fordert die Schülerinnen auf, die Abkürzung ihres Schulnamens zu programmieren. Als Hilfestellung bekommen sie schon ein Teil der Aufgabe programmiert. Anhand dieser Vorgabe sollen sie nun die Aufgabe lösen. Ein Betreuer sollte den Prozess beobachten und wenn nötig Hilfestellung geben. Abbildung 5 zeigt die Entwicklungsumgebung von LOGO.

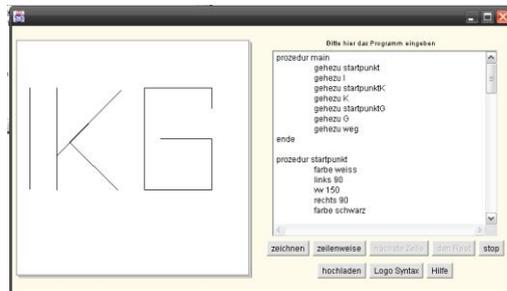


Abbildung 5: Entwicklungsumgebung LOGO. Ausgabefenster (links). Befehlseingabe (rechts).

Scratch⁵

Scratch ist eine kostenlose Programmierumgebung, bei der Code Teile per Drag&Drop zusammengefügt werden. Code Teile sind unter anderem:

- Schleifen
- Variablen
- Töne
- Bewegungen

Die Scratch-Programmieraufgabe ist die letzte Aufgabe der Schnitzeljagd. Um das Lösen zu erleichtern, können während des Spielabschnitts innerhalb von Gebäude 9 an verschiedenen Orten Code Teile aus Buntpapier eingesammelt werden.

Abbildung 6 zeigt die Lösung der Aufgabe.



Abbildung 6: Scratch-Aufgabe

3.3.5 EVALUATION DER AUFGABEN

Zum jetzigen Zeitpunkt konnte eine Evaluation der Aufgaben mit den Smartphones nicht durchgeführt werden, da die Aufgaben noch nicht in das System integriert sind. Deshalb erfolgte eine erste Evaluationsrunde mit einem Teil der Aufgaben der Außenroute in Papierform. Hierfür wurden zehn Probanden der Zielgruppe herangezogen. Diese wurden in zwei Gruppen zu je fünf Personen aufgeteilt. Jede Gruppe hatte die Zielvorgabe, die Aufgaben in Eigenregie zu lösen. Es wurden folgende Erkenntnisse daraus gewonnen:

- Wie viel Zeit brauchen die Schülerinnen für die jeweiligen Aufgaben.
- Sind die Aufgaben mit dem derzeitigen Bildungsstand lösbar.
- Gibt es Verständnisprobleme in den Erklärungen der Aufgaben.
- Haben die Schülerinnen Spaß an den Aufgaben.

Ebenso wurde das Gruppenverhalten beobachtet. Die Gruppen wurden getrennt voneinander evaluiert. In diesem Zeitraum bestand kein Kontakt der Gruppen zueinander. Die Ergebnisse bezüglich

⁵ <http://scratch.mit.edu/>

der Zeit sind in **Fehler! Verweisquelle konnte nicht gefunden werden.** dargestellt.

Tabelle 2: Ergebnisse bezüglich der Zeit

Aufgaben	Zeit Gruppe A in Minuten	Zeit Gruppe B in Minuten
A01	3:14:00	2:43:00
A02	4:12:00	3:56:00
A03	6:34:00	7:02:00
A04	2:23:00	2:04:00
A05	5:47:00	6:08:00
A06	4:38:00	5:25:00
A07	6:03:00	6:38:00

Den Aufgabenkriterien entsprechend (siehe Kapitel 3.3.2), wurden die Aufgaben auf 5-10 Minuten begrenzt. In **Fehler! Verweisquelle konnte nicht gefunden werden.** ist zu sehen, dass diese Vorgabe von beiden Gruppen nicht überschritten wurde. Somit müssen bezüglich des Zeitaufwands keine Änderungen der Aufgaben vorgenommen werden.

Alle Aufgaben konnten in der Gruppe gelöst werden. Demnach wurden die Aufgabenstellungen von den Probanden verstanden. Die einzelnen Mitglieder einer Gruppe halfen sich gegenseitig. Die Probanden haben die Aufgaben positiv aufgefasst und hatten Spaß am Lösen. Sie äußerten das Interesse, die Schnitzeljagd nach Fertigstellung mit den Smartphones zu wiederholen.

3.4 BELOHNUNGS- UND PUNKTESYSTEM

Die Spieler sammeln beim Lösen der Aufgaben Punkte in Form der Punkte des European Credit Transfer Systems (ECTS). Zum Lösen einer Aufgabe gibt es drei Versuche. Abhängig von der Anzahl der benötigten Versuche erhalten die Spieler mehr oder weniger Punkte. **Fehler! Verweisquelle konnte nicht gefunden werden.** zeigt die Punkteverteilung abhängig vom Lösungsversuch. Die Zahl in der Klammer gibt an, welche Punktzahl bei den normalen Aufgaben (normale Aufgaben = alle Aufgaben bis auf die letzte Aufgabe) gegeben wird. Bei der letzten Aufgabe können 100 Punkte erreicht werden.

Tabelle 3: Punktevergabe

	Versuch 1	Versuch 2	Versuch 3	Versuch 4
Punkte	Punkt-zahl (25) der Aufgabe * 3	Punkt-zahl (25) der Aufgabe * 2	Punkt-zahl (25) der Aufgabe * 1	Keine Punkte

Basis für die Punktzahlen, die erreicht werden können, sind die ECTS-Punktzahlen. Diese richten sich nach der Anzahl, die man zum Erreichen eines bestimmten Abschlusses im Studiengang Medien- und Kommunikationsinformatik an der Hochschule Reutlingen braucht. Für den Abschluss des Grundstudiums zum Beispiel müssen 60 ECTS-Punkte erreicht werden. Für den Masterabschluss sind einschließlich des Bachelorabschlusses 300 Punkte erforderlich. Für die Level „Doktorarbeit“, „Doktor“, „Hohes Tier bei eigener IT-Firma“ und „Welt-IT-Konzern“ wurden eigene, entsprechend höhere, Werte gewählt.

Tabelle 4 zeigt die Vergabe des Levels in Abhängigkeit der erreichten Punktzahl.

Tabelle 4: Erreichtes Level in Abhängigkeit der Punkte

Punkte	Erreichtes Level
0 - 30	Studienanfänger
31 - 60	Grundstudium abgeschlossen
61 - 120	Werksstudent
121 - 180	Bachelor abgeschlossen
181 - 240	Berufserfahrung gesammelt
241 - 300	Master abgeschlossen
301 - 600	Doktorarbeit
601 - 900	Doktor
901 - 999 Punkte	Hohes Tier bei eigener IT-Firma
1000 Punkte	Welt-IT-Konzern

Es gab die Überlegung, den Punkten, anstatt akademischer Grade, Karrieren zuzuordnen. Diese Idee wurde verworfen, da es diskriminierend wäre, eine Berufsgruppe als höher angesehen als andere darzustellen. Dennoch werden in Augmenture Karrieren, die mit dem Thema Informatik zu tun haben, an einem GPS-Punkt aufgezeigt.

3.5 AVATAR

Der Avatar basiert auf dem A des Augmenture-Schriftzugs (siehe Abbildung 7) und wurde in 3ds Max 10 modelliert. Er visualisiert den aktuellen Punktestand. In Abbildung werden der Punktestand „Studienanfänger“ (oben) und „Doktor“ (unten) dargestellt. Je nach Punktestand trägt der Avatar andere Accessoires. Im Falle des Studienfängers wäre es ein Buch und im Fall des Doktors der Doktorhut.

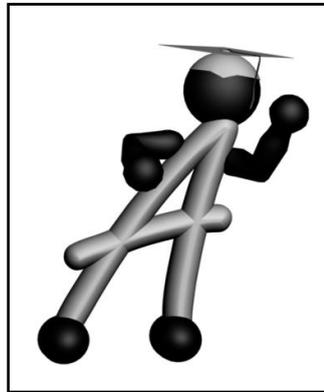
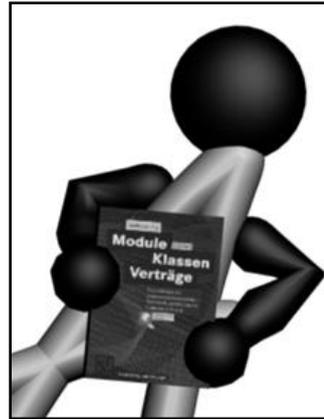


Abbildung 7: Augmee; Studienanfänger (oben), Doktor (unten)

4. FAZIT UND AUSBLICK

4.1 FAZIT

Dieser Artikel befasste sich mit dem Spiel „Augmenture“. Er beschrieb, wie bei der Konzeption und der Umsetzung vorgegangen wurde. Mit Augmenture wird eine Anwendung erstellt, die das Interesse am Thema Informatik wecken soll.

Es stellte sich heraus, dass die Aufgaben des Informatik-Biber-Wettbewerbs für eine virtuelle Schnitzeljagd geeignet sind. Sie lassen sich mit geringem Aufwand an eigene Bedürfnisse anpassen.

Das Projekt soll Ende April 2011 fertig gestellt sein und wird im Juli 2011 mit einer Schülergruppe evaluiert (siehe Kapitel 4.2).

4.2 GEPLANTE EVALUATION MIT SCHÜLERGRUPPE

In der Woche vom 18. – 22. Juli 2011 ist ein Spieldurchlauf mit Schülerinnen des Isolde-Kurz Gymnasiums in Reutlingen geplant. Laut unserem Kontakt zu diesem Gymnasium gibt es in der

Klassenstufe 9 25 Interessenten (Mädchen und Jungen) für das Spiel. Geplant war es, zwei Teams von je fünf Personen zeitgleich antreten zu lassen. Aufgrund des großen Interesses wäre es denkbar, die Veranstaltung entweder mehrmals an einem Tag durchzuführen, oder mehrere Tage für die Spiele einzuplanen.

Im Anschluss an das Spiel ist ein Gespräch mit den Teilnehmern geplant, um ihre Erfahrungen, die während des Spiels gemacht wurden, zu dokumentieren. Schwerpunkte werden hierbei sein, die Aufgaben mit Informatikthemen zu verbinden, zu erfahren, wo die Spieler Schwierigkeiten hatten und ob Augmenture Spaß macht und einen Wiederspielwert hat. Die Erkenntnisse aus dem Gespräch werden zur Verbesserung des Projekts verwendet.

5. LITERATURVERZEICHNIS

- [24] MINT-Quote unter Frauen Deutschland. <http://www.mintzukunftschaften.de/307.html>.
Letzter Zugriff: 10.04.2011
- [25] komm mach MINT. <http://www.komm-mach-mint.de/>. Letzter Zugriff, 11.04.2011
- [26] Kallenberg, P. April 2011. Programmierschnittstellen von Augmented Reality Browsern im Rahmen des Projekts „Augmenture“, Tagungsband Informatics Inside 2011
- [27] Schultheiß, R. April 2011. Mobile Augmented Reality auf Smartphones mit dem AR Browser Junaio, Tagungsband Informatics Inside 2011
- [28] Thiersch, B. April 2011. Entwurfsentscheidungen zur Augmented Reality Schnitzeljagd „Augmenture“, Tagungsband Informatics Inside 2011
- [29] Breuer, J., Bente, G. 2010. Why so serious? On the Relation of Serious Games and Learning, *Eludamos. Journal for Computer Game Culture*. 4 (1), p. 7-24
- [30] García-Crespo, Á., Colomo-Palacios, R., Gómez-Berbís, J. M. 2009. IT Professionals' Competences: High School Students' Views, *Journal of Information Technology Education*, Volume 8
- [31] Beyer, S., Rynes, K., Perrault, J., Hay, K., Haller, S. Februar 2003. Gender Differences in Computer Science Students, SIGCSE '03: Proceedings of the 34th SIGCSE technical symposium on Computer science education
- [32] Computer Science Unplugged. <http://csunplugged.org/>. Letzter Zugriff: 04.04.2011
- [33] Informatik-Biber. <http://www.informatik-biber.de/>. Letzter Zugriff: 05.04.2011
- [34] Pohl, W., Schlüter, K., Hein H-W. 2009. Informatik-Biber: Informatik-Einstieg und mehr
- [35] Hromkovič, J., 2010. Einführung in die Programmierung mit LOGO: Lehrbuch für Unterricht und Selbststudium. Vieweg+Teubner, 1. Auflage

XML3D Physik - Spezifikation einer Physikerweiterung und Integration einer Physik-Engine in einen XML3D-fähigen Browser

Fabian Liedtke
Hochschule Reutlingen

fabian_frederik.liedtke@student.reutlingen-university.de

ABSTRACT

In diesem Artikel wird eine Erweiterung der Auszeichnungssprache XML3D für 3D-Inhalte im Browser um die Simulation von physikalischen Kräften vorgestellt und eine Umsetzung mit der *Bullet Physics Engine* beschrieben.

1. EINLEITUNG

Ausgehend von der Anzeige von Webinhalten wie strukturierten Textdaten und Bildern in einem Browser, gibt es heutzutage eine Fülle von neuen Anwendungsgebieten. Bisher wurde vor allem die Umsetzung von 2D-Inhalten in den Vordergrund gestellt. Im Zuge der Standardisierung von HTML5 soll nun auch die dritte Dimension für Webanwendungen durch eine Erweiterung des Standards erschlossen werden. Durch das Zusammenwachsen der beiden Technologiefelder ergeben sich neue Möglichkeiten zur Darstellung von Webinhalten. So wird es möglich, bereits verfügbare Techniken aus 3D-Grafikanwendungen zu verwenden und in den neuen Webkontext einzubringen.

Im Folgenden wird die Auszeichnungssprache XML3D für 3D-Inhalte im Browser und die Funktionsweise aktueller Middleware zur Simulation von physikalischen Kräften vorgestellt. Eine mögliche Erweiterung der XML3D-Spezifikation um die Physikfunktionalität wird vorgestellt und eine Umsetzung mit einer dazu ausgewählten Physik-Engine beschrieben.

2. XML3D

XML3D wird derzeit als Gemeinschaftsprojekt des Deutschen Forschungszentrums für Künstliche Intelligenz, des Lehrstuhls für Computergrafik an der Universität des Saarlands und des Intel Visual Computing Institute entwickelt. Neben den W3C Standards HTML, CSS (Cascading Style Sheets) und DOM (Document Object Model), soll XML3D der Webstandard für die Darstellung von dreidimensionalen Inhalten in Webbrowsern werden. Die Spezifikation strebt eine hohe Kompatibilität mit den bekannten Dynamic HTML (DHTML) Technologien an. Dies soll einen einfachen Zugang zur

Erstellung von 3D-Inhalten ermöglichen [1]. Im Vergleich zu anderen Verfahren wie WebGL und O3D müssen keine neuen Datenmodelle, Skriptsprachen oder der Umgang mit neuen Programmierschnittstellen für die Integration von 3D-Inhalten in Webseiten erlernt werden. Die Anwendung von XML3D gleicht dem Umgang mit HTML und SVG (Scalable Vector Graphics) in der Verwendung von CSS, der Ereignisverarbeitung und dem Zugriff auf das DOM. Wie in der Auszeichnungssprache XML (Extended Markup Language), kann der XML3D Code direkt in ein XHTML Dokument eingebunden werden oder als externe Datei referenziert werden.

Verschiedene grafische Objekte, wie Dreiecksgitter, Linien und Punkte, lassen sich durch XML3D in einer 3D-Szene beschreiben. Diese Objekte lassen sich entsprechend dem XML Standard gruppieren, transformieren und stilisieren. Die Implementierung erlaubt die Verwendung von Raytracing- oder Rasterisierungsverfahren zur Darstellung der 3D-Inhalte. Außerdem können die für parallele Vektorberechnungen optimierten Grafikkprozessoren (GPUs) moderner Grafikkarten verwendet werden. Das Konzept der Vertex Arrays aus der OpenGL Spezifikation wird zur Organisation der grundlegenden 3D-Datenstrukturen (Dreiecksgitter) verwendet. Für die Darstellung von Oberflächeneigenschaften, wie zum Beispiel dem Glanz eines Metalls, können sogenannte Grafikkarten-Shader eingesetzt werden. Da für die Darstellung verschiedene Renderer eingesetzt werden können, wird die Shadersprache AnySL verwendet, die mit den gängigen Shadersprachen verschiedener Grafikkartenprogrammierschnittstellen kompatibel ist.

Durch die DOM-API lassen sich mit JavaScript Elemente oder Attribute hinzufügen, verändern und entfernen. So werden durch Manipulation der Dreiecksgitter zum Beispiel Animationen realisiert. Die DOM-Ereignisverarbeitung ermöglicht eine direkte Interaktion mit der 3D-Szene, wie zum Beispiel Drag and Drop von 3D-Objekten [1]. Codebeispiel 1 zeigt die Beschreibung eines Würfels nach der XML3D Spezifikation [2].

```

<html xmlns="http://www.w3.org/1999/xhtml">
<head> ... </head>
<body>
<h1>Cube Example</h1>
<xml3d
  xmlns="http://www.xml3d.org/2009/xml3d"
  style="width: 320px; height: 240px;">
  <defs>
    <transform id="boxTransform1"
      translation="0.1 7 5"/>
    <data type="triangles" id="cube">
      <int name="index">0 1 2 2 3 0...
    </int>
    <float3 name="position">-1.0 -1.0 -1.0
      -1.0 1.0 -1.0...</float3>
    <float3 name="normal">0.0 0.0 -1.0 0.0
      0.0 -1.0 ...</float3>
    <float2 name="texcoord">1.0 0.0 1.0
      1.0 0.0 1.0 ...</float2>
    </data>
  </defs>
  <group transform="#boxTransform1">
    <mesh src="#cube"/>
  </group> ...
</xml3d>
</body>
</html>

```

Codebeispiel 1 - XML3D Würfel Beschreibung

3. PHYSIK-ENGINES

Physik-Engines sind Middlewarelösungen für die Nachbildung von physikalischen Kräften. Im Vergleich zu wissenschaftlichen, numerisch genauen Simulationen, wie zum Beispiel in der Strömungslehre, sind Physik-Engines auf Echtzeitberechnungen ausgelegt und auf Performanz optimiert. Dabei steht das visuelle Ergebnis im Vordergrund, um zum Beispiel die Immersion in einem Computerspiel zu erhöhen. Es existieren mehrere frei verfügbare sowie kommerzielle Anwendungen. Für die Umsetzung des *XML3D Physics Plug-in* wurden verschiedene Engines verglichen und anhand ihrer Eigenschaften, Parallelisierbarkeit der Berechnungen, Lizenzkosten und unterstützter Plattformen eingeordnet. Aufgrund der Open-Source-Lizenz (Zlib) und dem erprobten Einsatz in Filmen und Computerspielen wurde die *Bullet Physics Engine* [4] für die Umsetzung ausgewählt.

Generell werden bei Physik-Engines verschiedene Arten der Physikberechnung unterschieden. Die *Bullet Physics Engine* unterstützt die Berechnung von Festkörperobjekten (*Rigid Body Dynamics*) und weichen, verformbaren Körpern wie Kleidung, Seilen und volumetrischen Körpern (*Soft Body Dynamics*). Als Festkörper werden Objekte verstanden, die ihre Form nicht verändern, weshalb aus Effizienzgründen für diese Objekte lediglich Translationen und Rotationen erlaubt sind. Für effiziente Kollisionsberechnungen zur Laufzeit der Simulation ist es außerdem notwendig den Objekten einer 3D-Szene abhängig von ihrem Verwendungszweck bestimmte Bewegungstypen (Motion Types) zuzuweisen. Bestimmte Objekte, deren räumliche Posi-

tion und Ausrichtung durch Kollisionen beeinflusst werden sollen, werden als dynamische Objekte angelegt. Wohingegen Objekte, wie zum Beispiel die Bodenebene einer Szene, als statische Objekte angelegt werden, da diese sich nicht bewegen sondern lediglich passiv mit den dynamischen Objekten kollidieren können.

Für die internen Kollisionsberechnungen werden eigene, durch Reduktion der Polygonzahl vereinfachte, geometrische Objekte verwendet, die auf der "realen", grafischen Repräsentation der Objekte basieren. Je nach Komplexität können speziell angepasste Formen verwendet werden. Prinzipiell kann die Physik-Engine mit Objekten mit hoher Polygonzahl umgehen, allerdings führt die explizite Deklaration von Objekten wie Kugeln, Quadern, komplexeren konkaven oder konvexen Objekten zu einer Performanzverbesserung. Eine Bowlingkugel muss zum Beispiel in der grafischen Repräsentation ein feines Dreiecksgitter aufweisen um "rund" zu erscheinen. In der Physikberechnung sollte für die Kugel jedoch die einfachere mathematische Beschreibung für die Kollisionsberechnung eingesetzt werden.

Ein Objekt besitzt in der Physik-Engine, ähnlich den Objekten in der echten Welt bestimmte physikalische Eigenschaften, die die Reaktionen bei Kraftereinwirkung beeinflussen. Dazu gehören Parameter wie Masse, Massezentrum, Reibung und Restitution. Die Restitution gibt dabei an, wie viel Bewegungsenergie bei Kollision für ein Objekt erhalten bleibt. Außerdem haben die Objekte eine bestimmte Position, Ausrichtung und Geschwindigkeit, die zur Laufzeit der Simulation durch Kollisionen mit anderen Objekten oder durch Einwirkung von Kräften dynamisch verändert wird.

Des Weiteren können Objekte bestimmten Beschränkungen unterliegen, die von der Physik-Engine gesondert behandelt werden müssen. Diese Beschränkungen werden als *Constraints* bezeichnet. *Constraints* dienen zum Beispiel der Nachbildung des Verhaltens einer Tür. Die Tür ist über ein Scharnier, der Türangel, mit einer Wand verbunden und kann sich deshalb nur bis zu einem gewissen Grad um eine Achse bewegen. In der Physik-Engine existieren mehrere verschiedene *Constraint*-Typen, welche die Freiheitsgrade der möglichen Bewegungen eines Objekts einschränken.

Ähnlich der Berechnungen von 3D-Grafik, wird auch in der Physik-Engine ein Pipeline-Berechnungsprinzip verwendet (siehe Abbildung 1). Die Objekte aus der 3D-Szene werden der Physik-Engine übergeben und gegebenenfalls für die Simulation vereinfacht. Im ersten Berechnungsschritt wird die Schwerkraft sowie die aktuelle Geschwindigkeit und so die damit verbundene

Translation und Rotation auf die betroffenen Objekte angewendet. Im zweiten Schritt werden Volumen für die Kollisionsdetektion der Objekte berechnet. Für Objekte bei denen eine Kollision auftritt, werden deren exakte Kontaktpunkte bestimmt. Im dritten Berechnungsschritt werden die sich aus den Kollisionen und *Constraints* ergebenden Bewegungseinschränkungen der Objekte angelegt und berechnet. Die neuen Koordinaten der Objekte werden übergeben und für die grafische Darstellung verwendet.

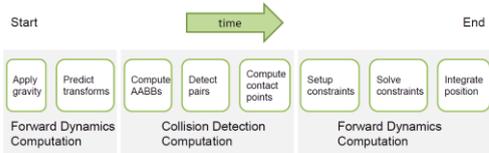


Abbildung 1 - Physikpipeline nach [3]

4. ERWEITERUNG DER XML3D-SPEZIFIKATION

Für das *Physics Plugin* wurde die XML3D Spezifikation erweitert, um physikalische Eigenschaften von 3D-Objekten beschreiben zu können. Dabei wird die Schwerkraft und weitere Parameter, wie die Simulationsgeschwindigkeit, global für die gesamte XML3D-Szene angegeben. Oberflächeneigenschaften, wie Restitution, Reibung aber auch der Bewegungstyp, werden dem bereits vorhandenen Material eines Objekts hinzugefügt. Die Masse eines Objekts wird direkt lokal bei der geometrischen Beschreibung eines Objekts definiert. Vereinfachte Objektformen, wie Kugel, Kegel oder Quader, können optional mit den entsprechenden Parametern zur Beschreibung angegeben werden. Werden keine angepassten Formen deklariert, wird lediglich das bereits angelegte Dreiecksgitter verwendet.

Einen Sonderfall stellen die *Constraints* dar, da diese jeweils die Verbindung zwischen zwei Objekten oder die Verbindung eines Objektes mit einem Punkt in der 3D-Szene darstellen. Für die *Constraints* ist es notwendig, die Objektkennung der zu verbindenden 3D-Objekte anzugeben. Codebeispiel 2 zeigt die erweiterten Objektbeschreibungen des Beispiels aus Codebeispiel 1 grau hinterlegt. Abbildung 2 zeigt das Ergebnis.

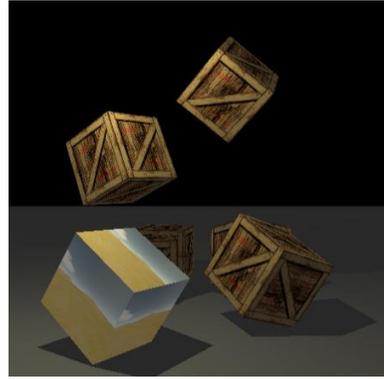


Abbildung 2 - Endergebnis im Browser

```

...
<xml3d
  xmlns="http://www.xml3d.org/2009/xml3d"
  style="width: 320px; height: 240px;" physics:gravity="0 -10 0">
  <defs>
    <transform id="boxTransform1"
      translation="0.1 7 5"/>
    <data type="triangles" id="cube">
      <float name="mass">10.0</float>
      <physics:shape>
        <string name="type">box</string>
        <float3 name="extends">2 2 2
        </float3>
      </physics:shape>
      <int name="index">0 1 2 2 3 0 ...
      </int>
      <float3 name="position">-1.0 -1.0 -1.0
        -1.0 1.0 -1.0 ...</float3>
      <float3 name="normal">0.0 0.0 -1.0 0.0
        0.0 -1.0 ...</float3>
      <float2 name="texcoord">1.0 0.0 1.0
        1.0 0.0 1.0 ...</float2>
    </data>
    <physics:material id="phCubemat">
      <string name="type">dynamic</string>
      <float name="friction">0.5</float>
      <float name="restitution">0.2</float>
    </physics:material> ...
  </defs>
  <group transform="#boxTransform1"
    physics:material="#phCubemat">
    <mesh src="#cube"/>
  </group> ...
</xml3d>
...
  
```

Codebeispiel 2 - Erweiterte XML3D Spezifikation

5. UMSETZUNG

Als Ausgangspunkt der Umsetzung wurden die native Implementierung von XML3D im Mozilla Browser Framework (Basis von Firefox) und WebKit (Basis von Google Chrome, Apple Safari, etc.) verwendet. Um die Erweiterbarkeit von XML3D zu untersuchen, sollten keine Änderungen am Quellcode der nativen Implementierung vorgenommen werden. Die Erweiterung und Physikberechnung wurde als Plug-in für Firefox und Chrome konzipiert. Dafür wurde die browserübergreifende Prog-

rammierschnittstelle NPAPI (Netscape Plug-in Application Programming Interface) verwendet, die Lese- und Schreibzugriffe auf das DOM eines XHTML Dokuments ermöglicht. Mit JavaScript werden alle Elemente aus dem Dokument ausgelesen und anhand der Bezeichnungen untersucht. Durch einen Parser werden alle notwendigen Informationen gesammelt und an die Physik-Engine zur Berechnung in geeigneter Form übergeben. Nach jedem Berechnungsschritt der Physik-Engine werden die Daten des DOM, die Translationen und Rotationen der 3D-Objekte, aktualisiert.

Die Interaktion mit Objekten einer Szene wird durch die DOM-Ereignisverarbeitung ermöglicht. Durch eine Ray-Cast-Funktion und die Daten der Betrachterkamera, können Objekte der 3D-Szene selektiert und manipuliert werden. Durch eine Schnittpunktberechnung des Sichtstrahls mit einem 3D-Objekt kann eine Kraft oder ein Impuls an der ermittelten Stelle ausgeübt werden. Abbildung 3 zeigt das Interaktionsmodell zwischen XML3D Browser und *Physics Plugin*.

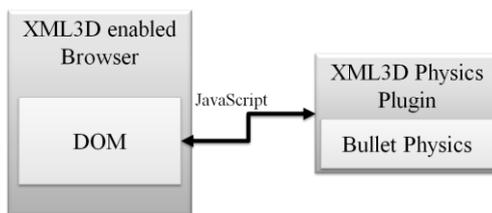


Abbildung 3 - Interaktionsmodell

6. FAZIT

Es konnte gezeigt werden, dass eine Erweiterung von XML3D durch ein Plug-in für die native Browser-Implementierung möglich ist. Trotz der aufwändigen Physikberechnungen konnten bei größeren Testszenen mit mehr als 300 dynamischen, kollidierenden Objekten selbst auf schwächeren Systemen akzeptable Ergebnisse erzielt werden. Mit dem Browserspiel "Gutter Balls" konnte ein

mögliches Anwendungsszenario für die Physikerweiterung aufgezeigt werden. Die Physik-Engine wird dabei für die Simulation einer Bowlingkugel, der Bahn und der Kegel eingesetzt. Mit den Mausereignissen der DOM-API kann auf die Kugel Einfluss genommen werden. Über JavaScript konnte auch ein einfaches Spielregelsystem entwickelt werden.

Eine aktuelle Version der nativen XML3D Implementierung von Firefox und das *Physics Plugin* sind unter [5] verfügbar.

7. LITERATURVERZEICHNIS

- [1] Sons, K., Klein, F., Rubinstein, D., Byelozorov, S., Slusallek, P. 2010. XML3D: interactive 3D graphics for the web. In *Proceedings of the 15th International Conference on Web 3D Technology (Web3D '10)*. ACM, New York, NY, USA, 175-184. DOI=10.1145/1836049.1836076 <http://doi.acm.org/10.1145/1836049.1836076>.
- [2] Sons, K., Klein, F., Rubinstein, D., Byelozorov, S., Slusallek, P., Brönnert, B., John, S. 2010. XML3D Spezifikation Version 0.4. Working Draft 20 Oktober 2010. Abrufbar unter: http://www.xml3d.org/wp-content/uploads/2010/10/xml3d_spec_0_4.pdf Letzter Zugriff: 21.03.2011.
- [3] Coumans, E. 2010. Optimizing Proximity Queries for CPU, SPU and GPU. SIGGRAPH 2010 Slides. Abrufbar unter: http://bullet.googlecode.com/files/SIGGRAPH_2010_Coumans_Erwin_OptimizingCD.pdf. Letzter Zugriff: 21.03.2011.
- [4] Bullet Game Physics. <http://bulletphysics.org/>. Letzter Zugriff: 21.03.2011.
- [5] Offizielle XML3D Internetseite. <http://www.xml3d.org/>. Letzter Zugriff: 21.03.2011.

Browserbasierte Multitouch-Interaktionen zur Steuerung von XML3D Szenen

Dirk Höntzsch

Deutsches Forschungszentrum für
Künstliche Intelligenz GmbH
dirk.hoentzsch@student.
reutlingen-university.de

Kai Stenkamp

Deutsches Forschungszentrum für
Künstliche Intelligenz GmbH
kai.stenkamp@student.
reutlingen-university.de

ABSTRACT

Dieser Artikel zeigt den aktuellen Stand von Interaktionen mittels Multitouch und untersucht im Speziellen die Multitouch-Unterstützung bei heutigen Browsern. Ferner wird ein universell einsetzbares Navigationskonzept für XML3D und dessen Umsetzung beschrieben. Dabei kann die Interaktion sowohl mit Single-Touch als auch mit Multitouch erfolgen.

1. EINLEITUNG

Die Entwicklung von Multitouch-Eingaben begann bereits vor mehr als 30 Jahren, jedoch wurde diese Art der Interaktion erst durch Apples iPhone populär [1]. Insbesondere die Multitouch-Unterstützung im Desktop-Browser ist ein Feld, das gerade an Bedeutung gewinnt. Hierzu werden die Erkenntnisse und Entwicklungen, die im Zuge der wissenschaftlichen Vertiefung gewonnen bzw. erarbeitet wurden, aufgezeigt. Die wissenschaftliche Vertiefung wurde am DFKI (Deutsches Forschungsinstitut für Künstliche Intelligenz) in Saarbrücken im Forschungsbereich Agenten und Simulierte Realität durchgeführt. Das Ziel bestand darin ein Konzept für ein Navigationsmodul zu entwickeln, das die Interaktion innerhalb XML3D Szenen übernimmt. Weiterhin sollte dieses Konzept implementiert werden. Dabei bestand die größte Hürde darin, die vom Browser geworfenen Touch Events entgegenzunehmen. Die Interaktion soll mittels direkten Multitouch-Eingaben im Browser erfolgen, dadurch soll eine Browserunabhängigkeit gewährleistet werden.

2. XML3D

XML3D wurde vom DFKI und dem Intel Visual Computing Institute an der Universität des Saarlandes unter der Leitung von Philipp Slusallek entwickelt. Mit XML3D können dreidimensionale Inhalte in XHTML (Extensible HyperText Markup Language) eingebunden werden. XML3D nutzt die Auszeichnungssprache XML (Extended Markup Language), wodurch die Objekte durch die Einbindung im DOM (Document Object Model) modifiziert werden können. Um realistische 3D-Szenen darzustellen, wird neben Rasterisierungsverfahren

auch Echtzeit-Raytracing unterstützt. Weiterhin können unter Verwendung von JavaScript und CSS (Cascading Style Sheets) Manipulationen an der 3D-Szene vorgenommen werden. [2]

Eine Einsatzmöglichkeit von XML3D zeigt die Abbildung 1:



Abbildung 1: XML3D Autokonfigurator

In dieser Anwendung sind unter anderem verschiedene Interaktionsmöglichkeiten mit dem Modell integriert, wie zum Beispiel das Öffnen und Schließen der Türen oder das Verändern der Beleuchtung.

3. MULTITOUCH

Um die Gebrauchstauglichkeit von Multitouch-Interaktionen bewerten zu können, müssen diverse Aspekte berücksichtigt werden. Es wird zwischen einhändiger und beidhändiger Interaktion unterschieden, wobei die beidhändige Interaktion noch um die *Multifinger*-Interaktion erweitert wird. In den meisten Fällen spricht man bei der beidhändigen Interaktion nur von einem Finger pro Hand. Des Weiteren spielen hierbei auch die zu erledigenden Aufgaben eine Rolle. Können beide Hände gleichzeitig zum Lösen der Aufgabe genutzt werden (z.B. zum Skalieren eines Objekts) oder über-

nimmt jede Hand eine separate Aufgabe (linke Hand zum Auswählen, rechte Hand zum Manipulieren der Objekte)? Die Effizienz ist aber jedes Mal von der Aufgabe abhängig und lässt sich nicht verallgemeinern. So schreiben Jiao et al., dass bei der Ausrichtung von Objekten eine Hand mit zwei Fingern bei Objekten ohne Ecken effizienter ist, während bei Objekten mit Ecken je ein Finger, aber beide Hände effizienter ist. [3]

Außerdem stellen Jiao et al. fest, dass beidhändige Interaktion die Parallelität steigern kann. [3] Weiterhin zeigen mehrere Studien, dass durch die Bedienung mit zwei Händen im Vergleich zur einhändigen eine deutliche Effizienzsteigerung erreicht wurde, wenngleich dies nicht verallgemeinert werden kann. [4] Zusätzlich wurde festgestellt, dass nur selten mehrere Finger gleichzeitig genutzt werden. Kin et al. kamen sogar zu dem Schluss, dass zu meist nur Zeige- und/oder Mittelfinger und niemals mehr als vier Finger gleichzeitig genutzt werden. [5]

Dennoch lässt sich bei der beidhändigen Benutzung aber auch eine erhöhte Fehlerrate feststellen, welche sich durch den Multifinger-Einsatz noch zusätzlich verstärkt. Kin et al. sehen unterem anderem die gesteigerte kognitive Last als eine Ursache für diesen Anstieg. [5] Es kann weiterhin zwischen direkter und indirekter Touch-Interaktion unterschieden werden, wobei bei ersterem die Ein- und Ausgabe auf ein und demselben Gerät stattfindet, in diesem Fall ein Multitouchdisplay.

Kin et al. schreiben, dass Multitouch im Idealfall doppelt so schnell wie eine Maussteuerung zum Lösen einer Aufgabe ist und dass es gerade für Anfänger keinen Unterschied macht, ob sie mit einem oder beliebig vielen Fingern arbeiten. [5]

3.1 ZWEIHÄNDIGES MULTITOUCH

Owen et al. [6] nennen drei entscheidende Vorteile von beidhändiger Multitouch-Interaktion:

1. Es führt zu einer messbaren Zeitersparnis.
2. Die Interaktionsform ist aus der realen Welt bekannt.
3. Es besteht die Möglichkeit schneller etwas zu manipulieren bzw. zu erkunden, was als *Expressiveness* bezeichnet wird.

3.2 SYMMETRISCHE / ASYMMETRISCHE AUFGABEN

Forschungen, insbesondere Guiards *Kinematic Chain* Modell [7], zeigen, dass die Hände zum größten Teil asymmetrisch bewegt werden. Das

heißt, dass die Aktion der nicht dominanten Hand der der dominanten Hand vorausgeht. Als anschauliches Beispiel wird das Schälen einer Kartoffel genannt: Zunächst hält die nicht dominante Hand die Kartoffel, erst danach wird sie mit der dominanten Hand geschält.

4. NAVIGATIONS-KONZEPTE

Ziel ist es, nur mittels Multitouch durch die 3D-Szene zu navigieren und somit auf den Einsatz von externen Geräten wie beispielsweise einen Datenhandschuh zu verzichten. Es gilt zudem, einen möglichst hohen Freiheitsgrad (degree-of-freedom, DOF) bei der Eingabe zu gewährleisten. Moscovich stellt einige 6-DOF-Controller vor, womit sich die drei räumlichen Achsen (z.B. zur Translation) und die drei winkligen Achsen (z.B. zur Rotation) steuern lassen. [8]

Diesen Grad an Freiheit gilt es auch bei der Touch-Interaktion sicherzustellen. Die Herausforderung ist insbesondere die Manipulation der dritten Achse. Mit Singletouch lässt sich maximal die X- und Y-Achse ansprechen, beispielsweise sind hierdurch nur Vorwärts- und Seitwärtsbewegungen, jedoch keine Aufwärtsbewegungen möglich.

5. NAVIGATIONS-KONZEPT FÜR XML3D

Aufbauend auf den zuvor erworbenen Kenntnissen zu Multitouch wurde für XML3D ein Navigationskonzept entworfen, welches nachfolgend dargestellt wird. Als Ausgangspunkt dient eine Interaktionsfläche, die einen Würfel enthält. Der Würfel steuert das Kameraobjekt der XML3D-Szene. Es ist am Mittelpunkt der Interaktionsfläche mittels eines virtuellen „Gummibandes“ befestigt. Dies bedeutet, dass der Würfel nach einer Bewegung in seine Ursprungsposition zurückgezogen wird. Das Verschieben des Würfels ermöglicht das völlig freie Bewegen in der Szene auf der Z- und X-Achse (Abbildung 2).

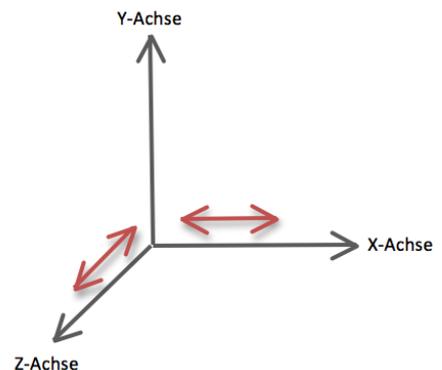


Abbildung 2: Koordinatensystem

Wird der Würfel in einer Position gehalten, setzt sich die Bewegung im Raum fort. Die Geschwindigkeit richtet sich hierbei nach der Entfernung vom Ursprung. Das heißt, dass auch durch nur geringes Verschieben des Würfels eine Positionsänderung hervorruft. Diese Technik ermöglicht ein freies und kontinuierliches Bewegen im Raum.

Eine Kameradrehung ist unter Verwendung einer Zwei-Finger-Geste möglich. Beim Auflegen von zwei Fingern auf den Interaktionsbereich wird ein Ring eingeblendet, der die Rotation der Kamera in eine beliebige Richtung der X-Achse steuert.

Abbildung 3 zeigt den Entwurf des Konzepts:

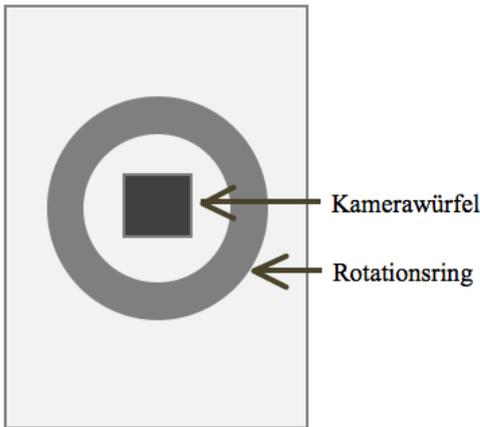


Abbildung 3: Konzept Navigation

Dieses Konzept soll universell einsetzbar sein, was bedeutet, es soll jedes Objekt in der 3D-Szene manipulieren können. Die Funktionalität lässt sich dabei an das ausgewählte Objekt anpassen, sodass statt einer Translation beispielsweise eine Skalierung resultiert. Ist kein 3D-Objekt ausgewählt, wird die Kamera gesteuert.

6. MULTITOUCH IM BROWSER

Bei der Entwicklung der Multitouch-Unterstützung wurde sich auf Firefox und WebKit Browser (im Speziellen Google Chrome) konzentriert, da für diese bereits native XML3D Implementierungen vorhanden sind. Opera wurde trotz experimenteller WebGL (Web Graphics Library) Unterstützung¹ nicht berücksichtigt, da es noch keine XML3D-fähige Version gibt und diese vorerst auch nicht geplant ist. Microsofts Internet Explorer bietet in seiner aktuellen Version 9 keine WebGL Implementierung, welche die Grundlage für XML3D darstellt.

¹ <http://my.opera.com/core/blog/2011/02/28/webgl-and-hardware-acceleration-2>

6.1 FIREFOX

Seit der Veröffentlichung der Beta 7 unterstützt Firefox 4.0 nativ Touch-Events. Sie sind identisch mit den Mausevents, besitzen jedoch zusätzlich das Attribut *streamId*, welches das eindeutige Identifizieren von mehreren Fingereingaben ermöglicht. Firefox unterstützt drei Arten von Touch-Events: *MozTouchDown* (wird ausgelöst sobald ein Finger den Touchscreen berührt), *MozTouchMove* (wird ausgelöst sobald ein Finger auf dem Touchscreen bewegt wird) und *MozTouchUp* (wird ausgelöst sobald ein Finger den Touchbereich verlässt).² Letzteres löscht die *streamId*, sodass diese für spätere Eingaben wiederverwendet werden kann. Codebeispiel 1 zeigt die Testimplementierung.

```
document.addEventListener("MozTouchDown",
mozillaTouchDown, false);
document.addEventListener("MozTouchMove",
mozillaTouchMove, false);
document.addEventListener("MozTouchUp", mozillaTouchUp,
false);

function mozillaTouchDown(e){
    e.preventDefault();
    console.log("down", e.streamId, e.clientX,
e.clientY);
}

function mozillaTouchMove(e){
    e.preventDefault();
    console.log("move", e.streamId, e.clientX,
e.clientY);
}

function mozillaTouchUp(e){
    e.preventDefault();
    console.log("up", e.streamId, e.clientX, e.clientY);
}
```

Codebeispiel 1: Implementierung Firefox

6.2 CHROME

WebKit Browser wie Google Chrome unterstützen mit *touchstart*, *touchmove* und *touchend* ähnliche Events.³ Die eindeutige Identifizierung der Finger lässt sich mittels der Eigenschaft „touches“ des *Touch*-Objekts des Events entnehmen.⁴ Erstmalige Verwendung fanden die Touch-Eingaben für WebKit in Apples Mobile Safari Browser. Gegenwärtig ist die Touch-Unterstützung nur in der mobilen Version vorhanden, was zum Ausschluss des Browsers führte.

² https://developer.mozilla.org/en/DOM/Touch_events

³ http://developer.apple.com/library/safari/#documentation/WebKit/Reference/TouchEventClassReference/TouchEvent/TouchEvent.html//apple_ref/doc/uid/TP40009358

⁴ http://developer.apple.com/library/safari/#documentation/appleapplications/reference/safariwebcontent/HandlingEvents/HandlingEvents.html//apple_ref/doc/uid/TP40006511-SW22

7. UMSETZUNG

Zu Beginn der Entwicklung wurde der Fokus auf Erweiterbarkeit gelegt. Ziel der Entwicklung ist es, ein Framework zu erstellen, welches flexibel um verschiedene Browser wie auch Geräte (Smartphone, Tablet Computer) erweitert werden kann.

7.1 BROWSER

Um eine browserunabhängige Lösung zu entwickeln, wurden verschiedene Tests mit den Browsern *Mozilla Firefox 4* und *Google Chrome* durchgeführt. Dabei stellte sich heraus, dass der auf WebKit basierende Browser Chrome sowohl in der Desktop- wie auch in der mobilen Version, welche bei dem Betriebssystem Android zum Einsatz kommt, keine nativen Touch-Eingaben unterstützt. Das heißt, die Events sind zwar implementiert, werden jedoch nicht ausgelöst, sodass auf diese nicht zugegriffen werden kann. Apple hingegen unterstützt mit den iOS-Geräten (iPhone, iPod Touch, iPad) auch im Browser native Touch-Eingaben. Für Mozilla Firefox ist in der Version 4.0 derzeit keine native Implementierung von XML3D vorhanden. Durch den Ausschluss des Google Chrome Browsers wird auf Mozilla Firefox und den JavaScript Renderer von XML3D zurückgegriffen. Durch dessen Verwendung können nahezu alle Browser, die eine WebGL-Unterstützung bieten, genutzt werden.

Zusätzlich wird zur Unterstützung fast aller Eingabegeräte eine Single-Touch-Implementierung vorgenommen, die auf herkömmliche Mousevents setzt. Dadurch ist auch die Nutzung der Maus als Eingabegerät möglich. Zum Einsatz kommt hier eine Browserweiche, welche zwischen Firefox und WebKit unterscheidet. Bei Firefox wird auf Übereinstimmung von Version 4.0 und Multitouch überprüft. Bei der Verwendung eines WebKit-Browsers wird auf iOS-Geräte getestet. Die Browserweiche schaltet die Multitouch Eingabemöglichkeit frei oder stellt auf Mouseingaben um.

7.2 ARCHITEKTUR

Als Programmiersprache dient JavaScript, welche von allen gängigen Browsern unterstützt wird. Um eine weitgehende Browserunabhängigkeit zu erreichen, kommt das Framework jQuery⁵ zum Einsatz. jQuery ist eine unter MIT oder GPL Lizenz veröffentlichte JavaScript Bibliothek. Diese stellt Funktionen zur komfortablen DOM-Manipulation und -Navigation zur Verfügung.

Eine Struktur bzw. ein Namensraum wird durch den Einsatz eines JavaScript Objekts, welches das gesamte Framework enthält, geschaffen. Abbildung 4 zeigt den Aufbau des Frameworks.

⁵ <http://jquery.com>

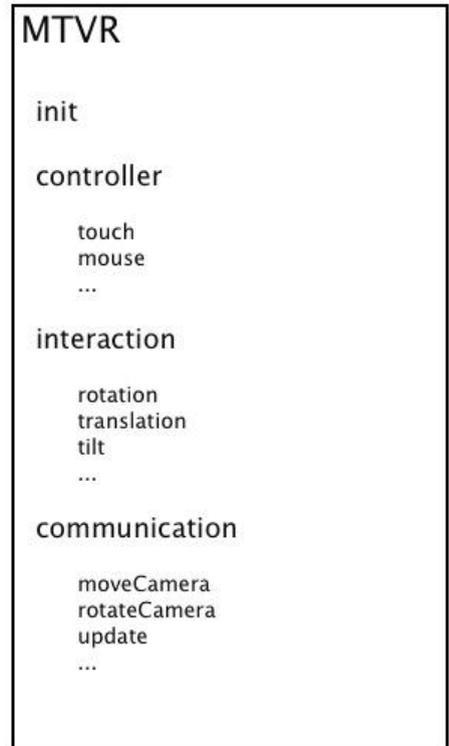


Abbildung 4: Architekturbild

- Bei der Initialisierung werden verschiedene Parameter gesetzt, zum Beispiel die Auflösung des Ausgabegeräts oder ob eine Touch-Unterstützung vorliegt.
- Der Controller ist für das Feststellen des Browsers und dessen Version zuständig. Weiterhin beinhaltet er die Touch- und Mousevent-Listener.
- Berechnungen wie beispielsweise die Distanz des Kamerawürfels zum Mittelpunkt werden im „interaction“-Teil des Frameworks vorgenommen.
- Die unter „communication“ zusammengefassten Funktionalitäten kommunizieren mit dem XML3D-Objekt. Es werden Bewegungen und Rotationen weitergeleitet und ausgeführt.

7.3 IMPLEMENTIERUNG

Zuerst wird überprüft, ob ein XML3D Objekt vorhanden ist. Sollte dies der Fall sein, wird die derzeit aktivierte View als Standard verwendet. Ausgehend von der Erweiterbarkeit wurde die im Abschnitt 7.1 erwähnte Browserweiche umgesetzt, welche die Überprüfung des verwendeten Browsers vornimmt. Es werden Mobile Safari und Firefox 4 unterschied-

den, bei allen anderen Browsern wird die Maus als Eingabegerät festgelegt.

Darauf erfolgt das Abfragen und Berechnen der Auflösung, die maximal im sichtbaren Bereich des Browsers dargestellt werden kann. Davon ausgehend werden die Interaktionsflächen in passender Größe erstellt, welche sich im Test wie folgt darstellen (Abbildung 5):

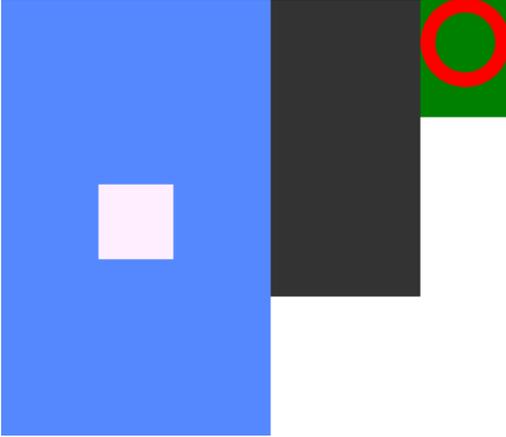


Abbildung 5: Interaktionsflächen

Zuletzt werden die vom Browser unterstützten Touch- oder Maus-Listener den Interaktionsschaltflächen hinzugefügt. Anschließend ist nach erfolgreichem Rendern das System einsatzbereit. Die folgende Abbildung (Abbildung 6) zeigt das System im Einsatz:

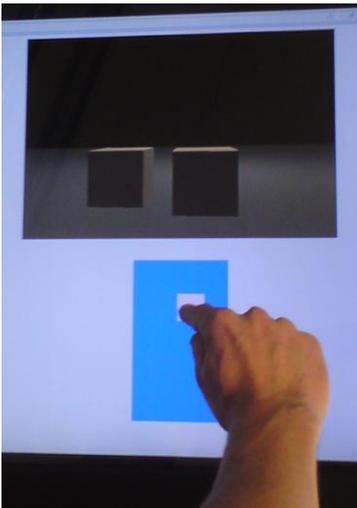


Abbildung 6: Framework im Einsatz

8. FAZIT

Die Interaktion durch Multitouchgeräte wird bereits seit längerem vorangetrieben, jedoch ist die Unterstützung im Browserbereich bisher eher vernachlässigt

worden. Hinzu kommt, dass es noch keine einheitlichen Standards hierfür gibt. Die dazugehörige W3C Spezifikation⁶ liegt lediglich als Entwurf vor und basiert zum größten Teil auf Apples iOS TouchEvent Spezifikation⁷.

Mozillas Multitouch-Implementierung in Firefox 4 funktioniert fehlerfrei und durch die Verwendung der *streamId* auch komfortabler. Jedoch ist fraglich, inwieweit sich diese Eigenentwicklung durchsetzt. Obwohl sich noch kein Standard etabliert hat, wird in der Praxis Apples Entwurf verwendet, da es sich auf allen WebKit-Browsern einsetzen lässt.

9. AUSBLICK

Wie beschrieben wurde bei der Umsetzung großer Wert auf einfache Erweiterbarkeit des Frameworks gelegt. Dadurch lassen sich leicht zusätzliche Geräte mit Multitouch-Unterstützung einbinden und um neue Interaktionsarten erweitern.

So ließen sich iOS oder Android Geräte beispielsweise für Präsentationszwecke nutzen. Denkbar ist auch die Einbindung von Bewegungssensoren, um damit zu navigieren und dadurch einen höheren Freiheitsgrad zu schaffen.

Angedacht ist auch die Steuerung über Gesten mit Hilfe von Microsofts Kinect⁸. Die Herausforderung ist hierbei in erster Linie eine geeignete Gestenerkennung. Die Interaktion ähnelt danach der der Multitouch-Umsetzung, sodass auf die bestehenden Navigationskonzepte zurückgegriffen werden kann.

Die Vorteile von Multitouch in Bezug auf die Steuerung von XML3D-Szenen und das erarbeitete Navigationskonzept werden durch eine Evaluation noch weiter untersucht.

⁶ <https://dvcs.w3.org/hg/webevents/raw-file/tip/touchevents.html>

⁷ http://developer.apple.com/library/safari/#documentation/UserExperience/Reference/TouchEventClassReference/TouchEvent/TouchEvent.html#//apple_ref/doc/uid/TP40009358

⁸ <http://www.xbox.com/de-DE/kinect/>

10. LITERATURVERZEICHNIS

- [1] Bill Buxton. 2007. Multi-touch systems that i have known and loved. <http://www.billbuxton.com/multitouchOverview.html>. Letzter Zugriff 19.04.2011
- [2] Kristian Sons, Felix Klein, Dmitri Rubinstein, Sergiy Byelozyorov, and Philipp Slusallek. Xml3d: interactive 3d graphics for the web. In *Proceedings of the 15th International Conference on Web 3D Technology, Web3D '10*, pages 175–184, New York, NY, USA, 2010. ACM.
- [3] Xiaodan Jiao, Hui Deng, and Feng Wang. An investigation of two-handed manipulation and related techniques in multi-touch interaction. In *Proceedings of the 2010 International Conference on Machine Vision and Human-machine Interface, MVHI '10*, pages 565–568, Washington, DC, USA, 2010. IEEE Computer Society. parallelism.
- [4] Paul Kabbash, William Buxton, and Abigail Sellen. Two-handed input in a compound task. In *Proceedings of the SIGCHI conference on Human factors in computing systems: celebrating interdependence*, CHI '94, pages 417–423, New York, NY, USA, 1994. ACM.
- [5] Kenrick Kin, Maneesh Agrawala, and Tony DeRose. Determining the benefits of direct-touch, bimanual, and multifinger input on a multitouch workstation. In *Proceedings of Graphics Interface 2009, GI '09*, pages 119–124, Toronto, Ont., Canada, Canada, 2009. Canadian Information Processing Society.
- [6] Russell Owen, Gordon Kurtenbach, George Fitzmaurice, Thomas Baudel, and Bill Buxton. When it gets more difficult, use both hands: exploring bimanual curve manipulation. In *Proceedings of Graphics Interface 2005, GI '05*, pages 17–24, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2005. Canadian Human-Computer Communications Society.
- [7] Yves Guiard. Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model, 1987.
- [8] Tomer Moscovich. *Principles and applications of multi-touch interaction*. PhD thesis, Brown University, 20

Kundenspezifische Anpassungen eines objektorientierten Systems unter der Verwendung des Service-Locator-Musters

Patrick Schuler
Hochschule Reutlingen

patrick.schuler@student.reutlingen-university.de

ABSTRACT

Dieser Beitrag zeigt, wie eine nach objektorientierten Prinzipien erstellte, betriebliche Software im Hinblick auf spätere, kundenspezifische Änderungen und Erweiterungen flexibel entworfen werden kann. Hierzu wird eine Kombination von bestehenden Entwurfsmustern vorgestellt.

1. EINLEITUNG

Im Rahmen der Weiterentwicklung, Änderung und Erweiterung der Funktionalität betrieblicher Anwendungssoftware stellt die Anpassung eines Service-basierten Software-Systems an veränderte Anforderungen des Kunden (*Customizing*) eine wesentliche Herausforderung für den Softwarearchitekten dar: Auch bei Veränderungen bestimmter Funktionalitäten und Geschäftsprozesse der Anwendung nach den Anforderungen des Kunden, muss ein stabiles Zusammenspiel zwischen bestehenden und neu zu entwickelnden Funktionen sichergestellt werden.

In einem nach objektorientierten Prinzipien gestalteten Anwendungssystem, kann die generelle Kompatibilität zwischen Standard- und kundenspezifischen Funktionen beispielsweise durch die Kapselung der jeweiligen Implementierung hinter definierten, gemeinsamen Programmschnittstellen realisiert werden: Eine abstrakte Klasse oder ein Interface definiert hierbei eine Menge fachlicher Operationen. Für jedes dieser Interfaces wird eine Standardimplementierung ausgeliefert. Basierend auf dem Prinzip der Trennung von Schnittstelle und Implementierung, erfolgt jeglicher Aufruf der Funktionalität im Code der Anwendung stets gegen den Typ der Schnittstelle und niemals gegen den Typ der eigentlichen implementierenden Klasse ([1] S. 240 ff.).

Soll eine Untermenge oder die gesamte Funktionalität der Standardimplementierung an die Bedürfnisse eines Kunden adaptiert werden, so kann im Rahmen des Customizings eine alternative, kundenspezifische Implementierung der gemeinsamen Schnittstelle bereitgestellt werden. Durch eine geeignete Konfiguration wird dann zur Laufzeit die kundenspezifische Implementierung anstelle der

Standard-Implementierung verwendet - für den aufrufenden Anwendungscode ist diese Änderung dank der Entkopplung von Schnittstelle und Implementierung und der Ausnutzung des Polymorphismus ([2] - S. 113 ff.) vollständig transparent.

Der vorliegende Beitrag beschreibt eine mögliche Umsetzung dieses Verfahrens auf Basis des *Service Locator* Musters ([3] - 9.3.1). Der nachfolgende Abschnitt definiert hierzu die grundlegenden Begriffe des *Service* und des *Service Locators*.

2. SERVICES UND SERVICE LOCATOR

Services dienen als primäre Bausteine einer Anwendung: Ein *Service* stellt eine definierte Funktionalität über ein Interface zur Verfügung, welche von anderen Diensten genutzt werden kann, um deren eigene Aufgaben zu erfüllen ([4] S.239 ff.).

In diesem Kontext wird ein *Service* konzeptionell in zwei Teile gegliedert:

In den *Service Type* welcher als Schnittstelle dient und die Semantik der Operationen die der *Service* zur Verfügung stellt definiert.

Der *Service Provider* stellt die konkrete Implementierung eines *Service Types* dar. Da ein Aufrufer den *Service Provider* nicht direkt instanziiert, sondern Instanzen des *Service Providers* nur durch den *Service Locator* ([3] - 9.3.1) erhält, bleibt dessen eigentlicher Typ und eventuell vorhandene Implementierungsdetails dem Aufrufer verborgen. Dies ermöglicht auch die Austauschbarkeit der Implementierung: Der Aufrufer nutzt nur die im *Service Type* definierten Operationen.

Der *Service Locator* verwaltet die *Service Provider*, deren Zugehörigkeit zum *Service Type* und instanziiert die angeforderten *Service Types*. Er ermöglicht die Trennung von Implementierung (*Service Provider*) und der Schnittstelle (*Service Type*) durch Kapselung der Instanziierung des *Service Providers*. Zusätzlich führt er zum einen das *Service Discovery* (Ermitteln der *Services*) und zum anderen den *Service Lookup* (Auffinden der *Services*) aus.

Diese Prozeduren werden in den nächsten Abschnitten näher erläutert.

Mögliche Anwendungsbeispiele für Services sind übergreifende Dienste wie Logging, Messaging, Internationalisierung oder sogar die komplette Repräsentation eines Dialogs durch einen Service.

Die Metainformationen zu den Services werden in diesem Szenario in einem so genannten *Service Binding* abgelegt. Hierbei werden die Metadaten: Service Type, Service Provider und *Scope* berücksichtigt.

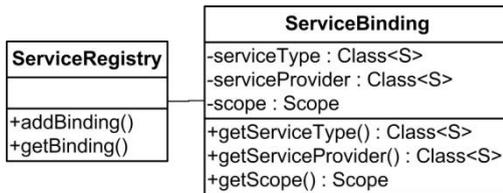


Abbildung 1 Service Registry & Service Binding

Mit dem Begriff *Scope* wird ein weiterer Bereich des Service Locators angesprochen. Zu seinen Aufgaben gehört unter anderem das Verwalten von Instanzen der verwendeten Service Provider.

Scopes sind assoziative Speicher, die der Service Locator benutzt, um die Abbildung eines Service Types auf eine konkrete Instanz des Service Types (den Service Provider) abzubilden.

Hierbei wird jedem dieser assoziativen Speicher eine definierte „Lebensdauer“ (Gültigkeit) zugewiesen. Die Instanzen in einem spezifischen *Scope* hängen direkt von der Lebensdauer des *Scopes* selbst ab: Endet dessen Gültigkeit, werden die darin liegende Instanzen verworfen.

Beispiele für einen solchen *Scope* sind ein *Application- und Window-Scope*.

Application-Scope:

Wird ein Service beispielsweise mit dem *Application-Scope* versehen, so wird bei der ersten Instanziierung des Service Types dieser vom Service Locator in einem dafür vorgesehenen assoziativen Speicher abgelegt. Die Gültigkeitsdauer dieses Speichers hält bis zum Schließen der Anwendung an. Kommt es zu einem erneuten Aufruf, wird der Service Type nicht neu erzeugt, sondern es wird die vorhandene Instanz aus dem assoziativen Speicher zurückgegeben. Dadurch wird sichergestellt, dass ein mit dem *Application-Scope* deklarierter Service nur einmal pro Anwendung instanziiert wird. Nachdem die Anwendung geschlossen wird, liert der assoziative Speicher seine Gültigkeit. Faktisch sind die Objekte im *Application-Scope* Singletons ([4] S. 157 ff.).

Window-Scope:

Bei Deklaration eines Services mit dem *Window-Scope*, wird der assoziative Speicher in dem die jeweiligen Instanzen abgelegt sind, mit dem Schließen des jeweiligen Fensters verworfen. Die „Lebensdauer“ der Instanzen beginnt und endet mit dem Öffnen und Schließen des Fensters.

Zur Verwaltung der Metadaten werden, wie in Abbildung 1 zu sehen ist, die Service Bindings in der Service Registry abgelegt. Diese dient als Registrierung für die Service Bindings und ist der zentrale Anlaufpunkt für den Service Lookup des Service Locators.

2.1 SERVICE DISCOVERY

Während des „Service Discovery“-Vorgangs liest der Service Locator einmalig beim Applikationsstart die Service Metadaten und speichert diese in Form von Service Bindings in der Service Registry ab. Dies kann zum einen über eine definierte Schnittstelle vom Entwickler erledigt werden, aber auch statisch über das Einlesen einer XML-Datei oder Datenbank. Eine weitere Möglichkeit ist die Verwendung von Code-Metadaten (Bsp.: Annotationen ([5] – 3.9) in Java). Sie ermöglichen es, Klassen direkt mit Metadaten zu versehen.

2.2 SERVICE LOOKUP

Das Auffinden eines konkreten Service Providers zu einem bestimmten Service Type geschieht über den sogenannten Service Lookup: Es handelt sich hierbei um eine definierte Folge von Schritten, anhand deren die jeweilige Instanz des konkreten Service Providers zu einem Service Type zurück gegeben wird.

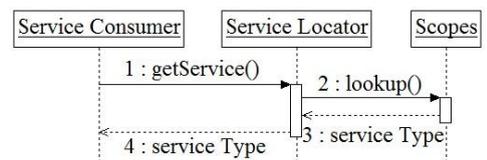


Abbildung 2 Service Lookup bei vorhandenem Scope

Wie in Abbildung 2 zu sehen ist, fragt der Service Consumer einen bestimmten Service Type beim Service Locator an (1), dieser wiederum durchsucht seine *Scopes* nach einer vorhandenen Instanz (2). Ist eine Instanz vorhanden, wird diese zurückgegeben (3). Dem Service Consumer steht der Service anschließend zur Verfügung (4).

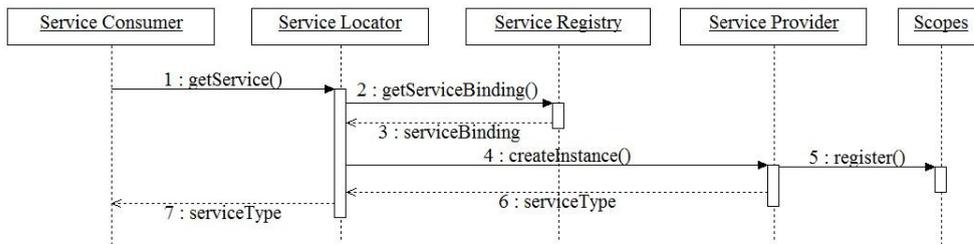


Abbildung 3 –Service Lookup ohne vorhandenem Scope

Abbildung 3 zeigt, dass bei der Anfrage eines Services (1), der noch nicht instanziiert wurde, dieser vom Service Locator über das Service Binding in der Service Registry gesucht wird (2).

Findet er einen passenden Service Provider (3), wird dieser instanziiert (4), dadurch wird die Instanz in dem, in den Metadaten definierten, Scope abgelegt (5). Die Instanz des Service Providers wird als konkrete Implementierung des Service Types durch den Service Locator an den Service Consumer übergeben (6 & 7).

3. CUSTOMIZING DURCH SERVICE METADATEN

Wie im vorigen Abschnitt beschrieben, ermöglicht der Einsatz des Service Locators die effektive Trennung und Entkopplung von Schnittstelle und Implementierung zur Laufzeit: Dies stellt bereits eine der notwendigen Voraussetzungen dar, um Standard-Implementierungen einer Schnittstelle transparent gegen kundenspezifische Implementierungen dieser Schnittstelle austauschen zu können.

Diese Eigenschaft für sich allein genommen, ermöglicht es jedoch noch nicht, eine qualifizierte Auswahl unter den zur Verfügung stehenden Service Providern eines Service Types zu treffen. Deshalb wird das Service Locator Muster um eine neue Eigenschaft erweitert: Diese wird als Precedence (dt. Präzedenz, Vorrang) eines Service Provider bezeichnet. Es handelt sich hierbei um eine zusätzliche Metainformation, die eine Rangordnung zwischen den vorhandenen Service Providern eines Service Types definiert. Die Berücksichtigung dieser Rangordnung muss als integraler Bestandteil des Service Lookups im Service Locator implementiert werden.

Für die konkrete Umsetzung dieser Eigenschaft sind mehrere Ansätze denkbar. Im Folgenden wird von einem pragmatischen Ansatz ausgegangen, in dem die Precedence in Form eines ganzzahligen Werts definiert wird: Der Service Locator wählt hierbei aus der Menge der zur Verfügung stehenden Service Provider zu einem Service Type immer denjenigen Service Provider aus, dessen Precedence den höchsten numerischen Wert besitzt.

Fordert ein Aufrufer nun eine Instanz eines Service Types vom Service Locator an, so ist dieser durch Berücksichtigung der Precedence in der Lage zwischen ggf. mehreren zur Verfügung stehenden Service Providern eine definierte Auswahl zu treffen. Soll nun die Standard-Implementierung einer bestimmten Funktionalität durch eine kundenspezifische Implementierung ausgetauscht werden, so ist es ausreichend, den kundenspezifischen Service Provider mit einer höheren Precedence als den Standard Service Provider zu versehen: Auf diesem Weg erhält der Aufrufer vom Service Provider als Resultat des Service Lookups die kundenspezifische Implementierung des Service Types, ohne weitere Codeanpassungen zur Nutzung der kundenspezifischen Funktionalität durchführen zu müssen.

Zur Veranschaulichung demonstriert der nächste Abschnitt den Einsatz dieses Musters anhand eines konkreten Beispiels.

4. ANWENDUNGSSZENARIO

In der Praxis ist folgendes Szenario denkbar:

Die Standardlösung einer Java-basierten, betrieblichen Anwendungssoftware soll beim Kunden eingerichtet werden. Standardmäßig wird die Anwendung mit einer SQL-Datenbank betrieben. Der Kunde wünscht allerdings eine NoSQL-Datenbank¹. Die in dieser Ausarbeitung behandelte Methodik bietet nun folgende Lösungsmöglichkeit: Das Interface welches den Datenbankzugriff ermöglicht, wird um eine zusätzliche Implementierung erweitert.

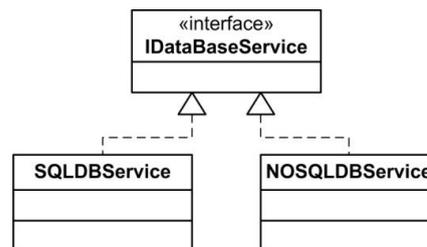


Abbildung 4 Aufbau des IDatabaseService

¹ <http://nosql-database.org/> (Zugriff: 24.03.11)

Der zuständige Entwickler fügt dem Projekt die Klasse *NOSQLDBService* hinzu. Diese implementiert, wie der standardmäßige *SQLDBService*, ebenfalls das Interface *IDataBaseService* (Abb. 4).

Für die Zuweisung der Metadaten werden in diesem Fall die von Java angebotenen Annotationen ([5] – 3.9) verwendet. Im Falle des Service Providers *SQLDBService* sieht diese Annotation folgendermaßen aus:

```
@ServiceProvider(serviceType=IDataBaseService.class, precedence=0, scope=Scope.APPLICATION)
public class SQLDBService implements IDataBaseService {
    //Implementierung der Zugriffsmethoden
}
```

In den Parametern der Service Provider-Annotation ist die konkrete Anwendung von Service Type, Precedence und des Scopes zu sehen. Der Service Provider wird aus dem konkreten Typ der annotierten Klasse abgeleitet.

Die Annotation trifft die Aussage, dass diese Klasse einen Service Provider zur Verfügung stellt: *@ServiceProvider(...)*. Der Service Provider ist vom Typ „*IDataBaseService*“, hat die Precedence „0“ und wird mit dem Application-Scope deklariert. Zu beachten ist hierbei, dass der Service Provider die im Interface „*IDataBaseService*“ bereit gestellten abstrakten Methoden implementiert.

Der ergänzte *NOSQLDBService* ist ebenso wie der *SQLDBService* vom Typ *IDataBaseService*, der entscheidende Unterschied der beiden Service Provider liegt in der Precedence und der Implementierungen der besagten Methoden. Der *NOSQLDBService* erhält eine höhere Precedence und die Implementierungen der Methoden sind für eine NoSQL-Datenbank ausgelegt.

```
@ServiceProvider(serviceType=IDataBaseService.class, precedence=1, scope=Scope.APPLICATION)
public class NOSQLDBService implements IDataBaseService {
    //Implementierung der Zugriffsmethoden
}
```

Der konkrete Aufruf des *IDataBaseService* (Abb. 5) über den Service Locator sieht im Quellcode wie folgt aus:

```
1: IServiceLocator serviceLocator = ...
2: IDataBaseService dbService =
serviceLocator.getService(IDataBaseService.class);
```

In Schritt 1 wird eine Instanz des Objekts vom Typ *IServiceLocator* erstellt. Mit Hilfe des Service Locators ist es möglich ein Objekt des Dienstes eines bestimmten Service Types, in diesem Fall *IDataBaseService*, anzufragen (2). Konnte die Instanz des Service Providers, der den Service Type implementiert, gefunden und somit zurückgegeben werden, stehen dem Service Consumer nun eine Instanz des Interfaces zur Verfügung. In diesem Fall gibt der Service Locator, aufgrund der höheren Precedence, die Implementierung des *NOSQLDBService* zurück. Somit ist es gelungen, die standardmäßige, betriebliche Anwendung, durch das Hinzufügen eines neuen Service Providers mit einer höheren Precedence, an die Kundenwünsche (Zugriff auf ein alternatives Datenbankmodell) anzupassen, ohne die Implementierung der bestehenden Anwendung zu manipulieren.

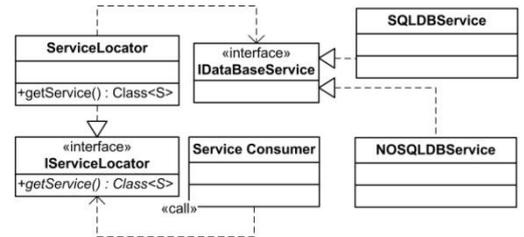


Abbildung 5 Aufruf des Service Locator

5. DANKSAGUNG

Bedanken möchte ich mich bei Herrn Harald Huber und Prof. Dr. Hertkorn für die hervorragende Betreuung.

6. LITERATURVERZEICHNIS

- [1] Oesterich Bernd. 1998. Objekt-orientierte Softwareentwicklung Oldenbourg
- [2] Silberbauer Christian. 2009. Einstieg in Java und OOP. Springer
- [3] Crawford, W., Kaplan, J. 2003. J2EE Design Patterns. O'Reilly
- [4] Gamma, E., Helm, R., Johnson, R., Vlissides, J. 2004. Entwurfsmuster, Elemente wiederverwendbarer objektorientierter Software. Addison-Wesley
- [5] Ullmann Christian. 2006. Java ist auch eine Insel. Galileo Computing

Ein Pattern-Katalog für REST-konforme Architekturen

Benjamin Blubacher
Hochschule Reutlingen
benjamin.blubacher@student.reutlingen-university.de

Frank Griesinger
Hochschule Reutlingen
frank.griesinger@student.reutlingen-university.de

Michael Haas
Hochschule Reutlingen
michael.haas@student.reutlingen-university.de

Ralph Kelkenberg
Hochschule Reutlingen
ralph.kelkenberg@student.reutlingen-university.de

Marita Klein
Hochschule Reutlingen
marita.klein@student.reutlingen-university.de

Prof.Dr.Alfred Zimmermann
Hochschule Reutlingen
alfred.zimmermann@reutlingen-university.de

ABSTRACT

Software is usually a system, which contains several parts to reach different targets. The division of these parts into smaller tasks shows that recurrent problems have to be solved. Design patterns represent in practice tested and proven solutions to such problems. This work describes an example scenario against which different design patterns for service-oriented architectures are presented. Based on REST and HTTP a media management system for home use is designed. As a result, the core areas of REST are revealed, problems in relation to these core areas and the example scenario are highlighted and proposed solutions in form of design patterns are described.

1. SZENARIO

Zu einfachen Personal Computern sind weitere Smart-Devices wie Notebooks, Smartphones, PDAs, TabletPCs oder Spielekonsolen im häuslichen Umfeld hinzugekommen [1].

Das Beispielszenario beschreibt eine verteilte Medienverwaltung im privaten Bereich. Dem Benutzer soll ermöglicht werden mithilfe von Smart-Devices über einen zentralen Server die Steuerung zu übernehmen. Drei verschiedene Arten der Teilnahme im System sind möglich. Geräte können als Steuerungseinheiten verwendet werden, mit der Möglichkeit Informationen einzusehen, Inhalte zu bearbeiten, Wiedergabe zu steuern. Darüber hinaus können Geräte Medieninhalte im Netzwerk zur Verfügung stellen. Informationen über die Medieninhalte werden vom zentralen Server gehalten. Die dritte Funktionalität, die ein Gerät besitzen kann, ist die Ausgabe von Medieninhalten, beispielsweise über Lautsprecher.

2. EINFÜHRUNG IN REST

Bei REST (Representational State Transfer) handelt es sich nicht um ein Protokoll, eine Technologie oder eine konkrete Architektur - es ist ein Architekturstil. Sandoval nennt es eine Menge von Bedingungen, welche einen Software-Architekturstil bilden, wenn sie beim Entwurf eingehalten werden (vgl. [San09], S. 7). Als Tim Burners-Lee die Grundlage für das World Wide Web erschaffen hatte, war seine Motivation: "create an easy-to-use, distributed, loosely coupled system for sharing documents" (s. [Web10], S.2). REST erstellte Fielding in seiner Dissertation (s. [3]), in dem er "von der konkreten Architektur [abstrahierte], die HTTP zugrunde liegt" (s. [4], S. 8). Dabei legte er den Fokus nicht auf die konkreten Protokolle oder ähnliches, sondern auf die Konzepte und Prinzipien. Der Erfolg der Architektur des Webs, welche mit HTTP realisiert wurde, sollte auch in anderen Web-Anwendungen, wie Web-Services, verwendet werden.

3. RESSOURCEN

Ressourcen sind das "zentrale Konzept von REST" (s. [4], S. 31). Bei REST-konformen (RESTful) Anwendungen sind die Funktionen auf die Ressourcen durch die Funktionalitäten des http-Protokolls beschrieben und universell zu sehen. Somit wird nicht in Funktionen, sondern in Ressourcen die Schnittstelle der Anwendung definiert. Unser System soll unter "<http://media-center>" zu erreichen sein. Einige entworfene Ressourcen sind in zu sehen.

4. REPRÄSENTATIONSFORMATE

Beim Nutzen des Webservices, welche bei REST durch das Bearbeiten von Ressourcen realisiert wird, werden die Repräsentationen der Ressourcen

verschickt, wodurch von der Implementierung und Verwaltung auf dem Server abstrahiert wird. Denn unabhängig davon kann der Client eine von ihm gewünschte Repräsentation der Ressource verlangen. Das "Alternative Format"-Pattern von Raj Balasubramanian et al. [5] ist ein Entwurfsmuster, welches von diesem REST-Prinzip inspiriert ist. Entwickler sollen ihre Dienste so entwerfen, dass sie Ressourcen in unterschiedlichen Formaten übertragen können.

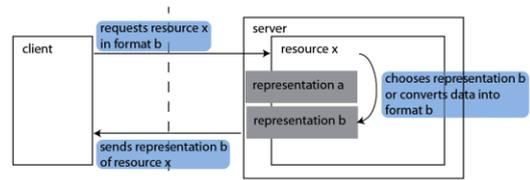


Abbildung 1: Visualisierung des "Alternate Format"-Patterns.

ENTWURFSMUSTER: ALTERNATIVE FORMATE

(Übersetzte Angaben von Raj Balasubramanian et al. auf [5].)

Problem:

Der Dienst-Konsument erfordert einen Datenaustausch, welcher nicht auf XML-Formaten basiert.

Lösung:

Dienste können derart entworfen werden, dass sie den Datenaustausch in alternativen Repräsentationsformaten vornehmen können, wodurch sie den Anforderungen spezieller Konsumenten entgegenkommen.

Erklärung:

Um alternative Formate zu unterstützen bieten sich zwei Ansätze an: Zum einen kann eine Technologie für Datenanhänge genutzt werden. "SOAP with Attachments" zum Beispiel ist ein Vorschlag des W3C zum Anhängen von nicht-XML-basierten Daten an SOAP Nachrichten. Auf dem Multipurpose Internet Mail Extension (MIME)-Standard aufbauend, werden im Header der Nachricht die Formate der einzelnen Nachrichtenteile beschrieben. Zum anderen können sich, durch die Nutzung von HTTP, Client und Server direkt auf ein Format einigen, in welchem die Ressource repräsentiert werden soll. In unserem Beispiel könnte ein Browser "text/html" als "Content-Type" im "Accept"-Feld des Headers der Anfrage angeben und eine Desktop-Anwendung würde etwa "application/vnd.example.standard+xml" anfragen.

Tabelle 1 Entwurf der wichtigen Ressourcen

Name	Adresse (http://media-center/)	Beschreibung	Typ
Devices	devices/	Liste der verfügbaren Geräte.	LR
Devices (einzeln)	devices/{id}	Gerät mit der bestimmten id.	PR
Playlisten	devices/{id}/playlists	Liste aller Playlisten auf einem Gerät.	LR
Playlist (einzeln)	devices/{id}/playlists/{id}	Playlist mit der bestimmten id.	SR
Media	media/	Liste mit den unterstützten Medientypen.	LR
Songs	media/songs/	Liste der Musikstücke.	LR
Songs (einzeln)	media/songs/{id}	Musikstück mit der bestimmten id.	PR
Filter	songs/?artist=dylan	Beispiel für die Anwendung eines Filters. Dies gibt alle Musikstücke an, welche im Attribut Künstler mit dem Suchbegriff "dylan" übereinstimmen.	F

5. HYPERMEDIA

Ein weiteres Grundprinzip von REST stellt die Verwendung von Hypermedia als Erweiterung vom Hypertext dar. Dieser Begriff betont den „strukturellen Aspekt, nämlich die nicht-lineare Informationsrepräsentation in einem Netz aus untereinander verknüpfter Knoten. Der Begriff Hypermedia [...] unterstreicht die multimediale Ausstattung dieses Netzes.“ ([6], S.23). Das Konzept der Links soll also Anwendungsübergreifend funktionieren und dank des globalen Namensschemas können so alle Ressourcen miteinander verknüpft werden. Nach Tilkov hat eine Webanwendung, die konform mit den REST-Prinzipien ist, daher folgende Eigenschaften (vgl. [4] S. 70):

- Links, also Verknüpfungen, bestimmen die Möglichkeiten des Clienten. Wie eine URI syntaktisch aufgebaut ist für den Clienten irrelevant.
- Es ist keine externe Beschreibung für die Verwendung der Anwendung notwendig. Die Präsentation ist gleichzeitig Dokumentation.
- Der Server verwendet ein für den Clienten verständliches Format, dass Hypermedia nutzt.
- Der Anwendungsstatus ist der serverseitige Ressourcenstatus.
- Server und Client sind entkoppelt.
- Änderungen an der Ressourcenverteilung sind transparent.
- Der Anwendungsablauf wird serverseitig gesteuert.

Hypermedia muss in der Interaktion von Client und Server eine zentrale Rolle spielen um eine REST-konforme Anwendung zu sein. Bei jeder Interaktion tauschen Service und Konsument Repräsentationen des Ressourcenzustandes, nicht des Applikationszustandes ([7], S.94).

Die Verlinkungen die für die Bereitstellung der Songs nötig sind zeigen auch hier die Vorteile von REST gegenüber SOAP. Möchte man eine Datei per Webservice versenden, muss das Format zuerst umgewandelt werden, da nur Byte, Integer oder Strings übertragen werden können. Nach der Übertragung muss dieses Format wieder zurückkonvertiert werden. Durch den Hypermedia-Ansatz in REST dagegen, können Links auf alle möglichen Medienarten gesetzt und direkt aufgerufen werden.

ENTWURFSMUSTER: ENTITY LINKING

In Verbindung mit diesem Ansatz wird nachfolgend das „Entity Linking“-Pattern erläutert. (*Übersetzte Angaben von Raj Balasubramanian et al. auf [8].*)

Problem:

Zwischen konzeptionellen Business-Entities besteht für gewöhnlich eine Verbindung. Services sind aber häufig autonom, also ohne Angaben über diese Verbindungen. Dienst-Konsumenten und die Steuerung der Services verfügen möglicherweise nicht über ausreichend Informationen, um diese Zusammenhänge herzustellen.

Lösung:

Services beinhalten Entity-Referenzen in den Servicefähigkeiten der Antwortnachricht.

Erklärung:

Antwortnachrichten von Ressourcen, die in Bezug zu einer Entität stehen, enthalten Referenzen zu diesen Entitäten. Diese Kennungen können auf Entitäten im selben, aber auch auf Entitäten in anderen Services referenzieren. Dieses Prinzip wird durch den beschriebenen Hypermedia Ansatz vollständig umgesetzt. Die Antworten der Ressourcen liefern in ihrem Rückgabewert Links, die zu entsprechend zugehörigen Ressourcen referenzieren. Lässt man sich beispielsweise eine Liste der Devices ausgeben (*GET http://media-center/devices/HTTP/1.1*) erhält man eine Liste mit verschiedenen Links, die wiederum zu den Ressourcen der einzelnen Devices führen. Auch kann über die Song-Liste direkt zu den MP3-Dateien referenziert werden. Etwas schwieriger ist es, den Verweis auf En-

titäten im selben Service in einer REST-konformen Umgebung umzusetzen. Da REST per se nicht Service-orientiert ist, lässt sich ein einzelner Service nicht so konkret abstecken. Der Service „Musik abspielen“ wird beispielsweise nicht als solcher angeboten, sondern durch das Ändern eines Attributes der Ressource „Playlist“ umgesetzt (z.B.: *currentSong=1828*). Demnach kann der Verweis auf verschiedene Ressourcen, je nach Anwendungsfall sowohl die Bedingung des Referenzierens auf Entitäten im eigenen, als auch in anderen Services erfüllen.

6. SITZUNG UND SKALIERBARKEIT

„Der Verzicht auf Sitzungsstatus ist ein in allen verteilten Systemarchitekturen angestrebtes Ziel.“ ([4], S.112) Der von REST verwendete Architekturstil beinhaltet eine Einschränkung bezüglich des Sitzungsstatus. Das hat den Vorteil das Eigenschaften wie Sichtbarkeit, Skalierbarkeit und Ausfallsicherheit verbessert werden (vgl. [4], S. 112).

Ein klarer Vorteil für den Server ist, dass Ressourcen persistent und identifizierbar sind. Die Auswirkungen sind, dass die Kommunikation explizit ist und jede einzelne Nachricht, ohne die vorhergehende zu kennen, verstanden werden muss. Die Request-Verarbeitung kann nun auf verschiedenen Servern stattfinden da aufeinander folgende Nachrichten nicht mehr von einem System bearbeitet werden müssen und es werden weniger Systemressourcen benötigt. Status behaftete Kommunikation sollte in REST-Anwendungen vermieden werden. Das erfolgt in dem der Status in einen Client- oder Ressourcenstatus umgewandelt wird (vgl. [4], S. 108).

Je nach Anforderung muss entschieden werden welcher der beiden Wege am sinnvollsten ist. Bei der clientseitigen Verwaltung des Status muss der Client bei jeder Interaktion mit dem Server seinen Status mit übertragen. Anhand dieser Daten kann der Server dann antworten. Probleme hierbei sind, dass die Menge der einzelnen Anfragen und Antworten enorm steigt. Ein weiteres Problem ist, dass der Server den aktuellen Status des Clients nicht sieht und somit nicht weiß, ob der Client momentan erreichbar ist (vgl. [4], S. 109).

Als Alternative kann auch ein Ressourcenstatus verwendet werden. Genau wie bei dem vorherigen Beispiel gibt es einen Server- und einen Clientstatus. Ein Nachteil dieser Variante ist, dass der Server nicht nur seinen eigenen Status sondern auch den der Clients verwalten muss. Die Konsequenz aus diesem Ansatz ist, dass keine Sitzungen mehr

benötigt werden und damit eine erhöhte Skalierbarkeit vorhanden ist (vgl. [4], S. 110).

7. CACHING

„Nichts ist effizienter als Anfragen, die gar nicht erst gestellt werden“ ([4] S. 113). Die Zwischenspeicherung von Informationen trägt zur Reduktion der zu übertragenden Datenmengen und gestellter Anfragen bei. Caches stellen Puffer zwischen anzeigendem Medium und Datenquelle dar. Bei wiederholtem Bedarf von Clients an einer Ressource kann auf die zwischengespeicherte Variante zurückgegriffen werden. Puffer ermöglichen die Entlastung der ursprünglichen Datenquelle. Außerdem wird die Zugriffszeit durch die Reduktion der Hops reduziert. Befindet sich der Cache nahe dem Client besteht zwischen beiden in der Regel eine höhere Bandbreite als zwischen dem Ursprungsserver und des Clients, was eine weitere Beschleunigung bei Zugriffen zufolge hat.

ENTWURFSMUSTER: TRANSPORT CACHING

(Übersetzte Angaben von Raj Balasubramanian et al. auf [9].)

Problem:

Das wiederholte Übertragen und Bearbeiten gleichbleibender Inhalte ist Verschwendung von Ressourcen. Der Server muss häufig angefragte Daten im Speicher halten und Anfragen wiederholt bearbeiten, das bedeutet eine höhere Hardwareauslastung auf Serverseite. Dies hat direkte Auswirkungen auf die Skalierbarkeit des Systems, da die Daten für jeden aktiven Client serverseitig gehalten und redundante Bearbeitungsschritte wiederholt ausgeführt werden. In Bezug auf das Beispielszenario müssten sämtliche aktiv genutzten Mediendateien sich im Speicher des Servers befinden.

Lösung:

Die Lösung nach [9] ist die Nutzung eines Kommunikations-Frameworks mit integrierten Caching-Funktionalitäten. Diese würden das Auslagern von Daten bis zur nächsten Nutzung ermöglichen.

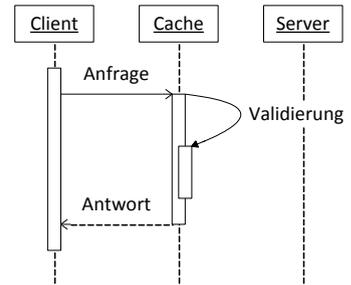
Vorteil: Daraus resultiert eine verringerte Datenmenge, die zu übertragen ist und vom Server im Hauptspeicher gehalten werden muss.

Nachteil: Für die Caches ist eine erweiterte Infrastruktur notwendig.

Heutige Architekturen nutzen für dieses Pattern gewöhnlich die Funktionalitäten,

die von HTTP zur Verfügung gestellt werden.

Erklärung:



Eine bearbeitete Anfrage an den Server wird zwischengespeichert, mit Informationen über die Feststellung der Aktualität der Daten. Erfolgt eine gleiche Anfrage erneut, wird überprüft ob das gepufferte Ergebnis den Anforderungen entspricht und wenn möglich als Antwort weitergeleitet. Dadurch wird eine erneute Berechnung und Verarbeitung der Daten von Seiten des Servers umgangen.

8. SICHERHEIT

Sicherheit ist bei jedem System wichtig, so ist in den meisten REST-Implementationen eine Kombination von HTTP Basic Authentication und SSL im Einsatz. Dadurch wird eine sichere Übertragung und Authentifizierung sichergestellt. Trotzdem können diese Nachrichten mit schädlichem oder fehlerhaftem Inhalt versehen sein. Das Message-Screening-Pattern befasst sich mit dieser Problematik. Wenn ein Dienst eine Nachricht empfängt, wird mit einer Reihe von Kontrollen der Nachrichteninhalt nach schädlichen Daten durchsucht. Eine sogenannte „Screening-Logik“ erfordert zusätzliche Routinen, um binäre Inhalte und Anhänge von Nachrichten zu bearbeiten. (vgl. [SOA10c]).

ENTWURFSMUSTER: MESSAGE-SCREENING

(Übersetzte Angaben von Raj Balasubramanian et al. auf [5].)

Problem:

Ein Angreifer kann Nachrichten mit schädlichem oder fehlerhaftem Inhalt zu einem Dienst übertragen, was zu unerwünschtem Verhalten führt.

Lösung:

Der Service ist ausgestattet oder ergänzt mit speziellen Screening-Routinen. Alle eingegebenen Daten sind zuerst einmal schädlich bis zum Beweis des Gegenteils.

Erklärung:

Sicherheit ist bei jedem System wichtig, so ist in den meisten REST-Implementationen eine Kombination von HTTP Basic Authentication und SSL im Einsatz. Dadurch wird eine sichere Übertragung sowie die Authentifizierung sichergestellt. Trotzdem können diese Nachrichten mit schädlichem oder fehlerhaftem Inhalt versehen sein.

Wenn ein Dienst eine Nachricht empfängt, wird mit einer Reihe von Kontrollen der Nachrichteninhalte nach schädlichen Daten durchsucht. Eine so genannte „Screening-Logik“ erfordert zusätzliche Routinen, um binäre Inhalte und Anhänge von Nachrichten zu bearbeiten. Trotz genauer Routinen ist es nicht möglich alle Formen von schädlichen Inhalten zu überprüfen. Je höher der Nachrichten Austausch ist desto höher wird auch die zusätzliche Laufzeit-Bearbeitung.

9. FAZIT

Der ressourcenbasierte Ansatz eignet sich gut für die Darstellung der einzelnen Systemelemente, jedoch ist REST ungeeignet für Echtzeitanwendungen. Ressourcen können aktualisiert, bzw. verändert werden aber die Verarbeitung vorhandener Informationen kann nicht über die Grenzen eines Teilsystems hinaus gesteuert werden.

10. LITERATURVERZEICHNIS

[1] Microsoft: Smart Devices –
[[\[de/library/cc431224.aspx\]\(http://msdn.microsoft.com/de/library/cc431224.aspx\)\], Einsichtnahme:
10.01.2011](http://msdn.microsoft.com/de-</p></div><div data-bbox=)

[2] Sandoval, Jose: *RESTful Java Web Services* - UK, Birmingham: Packt Publishing, 2009.

[3] Fielding, Roy Thomas: *PhD dissertation - Architectural Styles and the Design of Network-based Software Architectures* -
[http://www.ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation.pdf] - University of California, 2000.

[4] Tilkov, Stefan: *REST und HTTP - Einsatz der Architektur des Web für Integrationsszenarien* - 1. Auflage - dpunkt.verlag GmbH, 2009.

[5] *SOA Patterns: Alternative Format* -
[http://www.soapatterns.org/alternative_format.php], Einsichtnahme: 15.01.2011.

[6] Unz, Dagmar: *Lernen mit Hypertext: Informationssuche und Navigation* – Münster: Waxmann, 2000

[7] Webber, Jim; Parastatidis, Savas; Robinson, Ian: *REST In Practice* – Sebastopol: O’Reilly Media, 2010

[8] *SOA Patterns: Entity Linking* -
[http://www.soapatterns.org/entity_linking.php], Einsichtnahme: 12.01.2011.

[9] *SOA Systems Inc: Transport Caching* –
[http://www.soapatterns.org/transport_caching.php], Einsichtnahme: 15.01.2011

Einsatz von Graphenalgorithmen zur Planung des Zuspiels im Roboterfußball

Hannes Bergler
Fakultät für Informatik und Ma-
thematik
Hochschule München

Daniel Metzger
Fakultät Technik
Hochschule Reutlingen

Ursula Voß
Fakultät Technik
Hochschule Reutlingen
ursula.voss@reutlingen-
university.de

ABSTRACT

Der Studienbereich Mechatronik der Hochschule Reutlingen engagiert sich seit 2009 mit großem Erfolg in der Mixed Reality Liga des RoboCups, einer Initiative zur Förderung der Forschung und interdisziplinären Ausbildung in den Bereichen Künstliche Intelligenz und autonome mobile Systeme. In der Mixed-Reality-Liga spielen reale Miniroboter auf einem virtuellen Spielfeld mit einem virtuellen Ball Fußball. Dazu muss neben anderen Aufgaben eine künstliche Intelligenz entwickelt werden, nach der die fußballspielenden Miniroboter ihre Entscheidungen treffen. Eine Aufgabe ist die Realisierung eines sinnvollen Zuspiels zwischen den Spielern. Hier wird beschrieben, wie ein klassischer Graphenalgorithmus, der Dijkstra-Algorithmus, zur Planung einer Zuspielstrategie in diesem Multiagentensystem eingesetzt wird.

1. EINLEITUNG

RoboCup ist eine internationale Initiative zur Förderung der Forschung und interdisziplinären Ausbildung in den Bereichen Künstliche Intelligenz und autonome mobile Systeme [1].

„Fußball spielen“ ist aus dem Alltag bekannt. Es verlangt das Agieren in einer dynamischen Umgebung und die Integration der Handlungen des einzelnen in die Ziele der eigenen Mannschaft. Für den Roboterfußball müssen die Entwickler jeder Mannschaft geeignete Strategien entwickeln und implementieren.

2. DAS MIXED-REALITY-SYSTEM

Die RT Lions, das Team der Hochschule Reutlingen, spielen seit 2009 in der Mixed-Reality-Liga mit. In der Mixed-Reality-Liga spielen reale Miniroboter auf einem virtuellen Spielfeld mit einem virtuellen Ball (Abb. 1).



Abbildung 1: Spielszene in der Mixed-Reality-Liga

Das Mixed-Reality-System besteht aus dem komplexen Zusammenspiel diverser Hard- und Softwarekomponenten (Abb. 2).

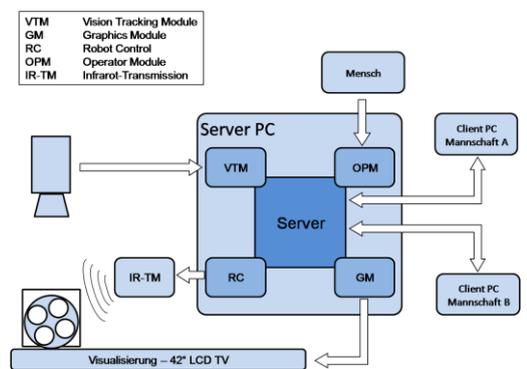


Abbildung 2: Komponenten des Mixed-Reality-Systems [2]

Auf einem liegend montierten 42-Zoll-Bildschirm spielen je fünf, beziehungsweise seit 2011 sieben Miniroboter gegeneinander.

Auf diesem Bildschirm werden das Spielfeld und der Ball visualisiert. Die 25x25x30 mm großen Miniroboter der Firma Citizen haben zwei Räder, und erhalten ihre Fahrbefehle in Form von Radgeschwindigkeiten, die über Infrarot übertragen werden. Da die Räder unabhängig angesteuert werden können, sind die Miniroboter sehr wendig.

Das Spielfeld wird von einer Kamera beobachtet. Die Roboter haben aufgeklebte Markierungen, an Hand derer sie eindeutig identifizierbar sind. Die Positionen der Roboter werden an den zentralen Server weitergeleitet. Dieser gibt zum einen die Positionen der Roboter an die Client-Rechner der beiden Mannschaften weiter, zum anderen erzeugt er die virtuellen Teile der Spielwelt. Nach physikalischen Gesetzmäßigkeiten werden die auftretenden Kräfte bei Kollisionen zwischen den virtuellen Repräsentationen der Roboter und die Ballbewegung bestimmt.

Innerhalb des Clients laufen Softwareagenten, von denen jeder die Steuerung genau eines Roboters übernimmt. Die Agenten eines Clients dürfen untereinander nicht kommunizieren. Jeder Agent ist Zentrum seines eigenen Polarkoordinatensystems. Die Positionen der Mitspieler, der Gegner, des Balls und der wichtigsten Punkte des Spielfelds sind jedem Spieler in Form von Vektoren bekannt.

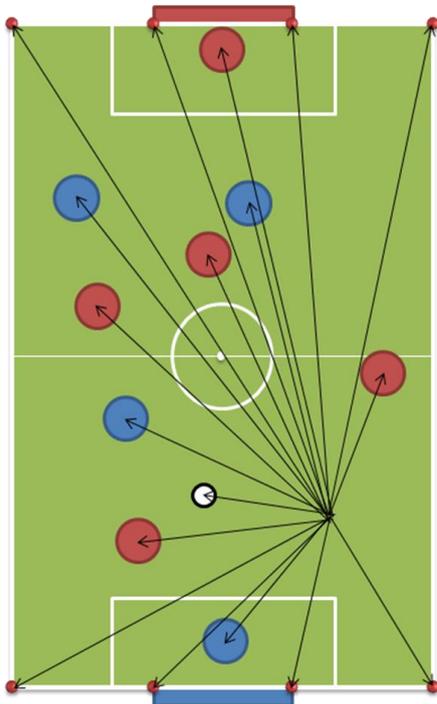


Abbildung 3: Alle einem Agenten bekannten Vektoren

3. DIE KÜNSTLICHE INTELLIGENZ (KI)

3.1 REAKTIVE SOFTWARE-AGENTEN

Vor diesen technischen Randbedingungen gilt es, die spielerische Intelligenz der Miniroboter zu entwickeln. Die Spieler sind reaktive Softwareagenten, d.h. sie verfügen nicht über ein Modell der Realität, sondern sie handeln nach Regeln, die einer wahrgenommenen Umweltsituation unmittelbar eine Reaktion folgen lassen. Ziel ist es, die Reaktionen der einzelnen Agenten so zu steuern, dass sie gemeinsam als Multiagentensystem ein gestelltes Problem lösen [3]. Übertragen auf das Fußballspiel besteht das gemeinsame Ziel darin, in Ballbesitz zu kommen und den Ball in das Tor der gegnerischen Mannschaft zu spielen.

3.2 EIN EINFACHER ANSATZ

Ein erster Ansatz sieht vor, dass ein Spieler, der nicht im Ballbesitz ist, sich zum Ball bewegt. Ist ein Spieler im Ballbesitz, schießt er in Richtung gegnerisches Tor. Dieses sehr einfache Regelwerk führt sehr schnell zu Verklemmungen und ist wenig effizient. Eine weitere Regel weist den Spielern wie im realen Fußball Positionen auf dem Spielfeld zu, von denen sie sich nur eine gewisse Distanz entfernen dürfen. Damit wird gewährleistet, dass die Spieler der Mannschaft gleichmäßig über das Spielfeld verteilt bleiben. Ist ein Spieler im Ballbesitz, leitet sich sein Verhalten daraus ab, wo auf dem Spielfeld er sich befindet: Aus der vorderen Spielhälfte wird direkt auf das Tor geschossen, unabhängig davon, ob ein Gegner in der Schusslinie steht oder nicht. Aus der hinteren Spielhälfte wird in Richtung definierter Stürmerpositionen geschossen, unabhängig davon, ob tatsächlich ein Mitspieler dort wartet [4]. Diese Spiellogik führt dazu, dass immer über die Flanken gespielt wird.

3.3 WEGEPLANUNG MIT DEM DIJKSTRA-ALGORITHMUS

Zur Verbesserung des Spielverhaltens wurden Ansätze aus der Graphentheorie verwendet [5]. Aus der Graphentheorie sind Algorithmen bekannt, mit denen man effizient den besten Weg von einem Startpunkt zu einem Zielpunkt bestimmen kann.

Übertragen auf die vorliegende Situation stellen die Mitspieler und das gegnerische Tor die Knoten eines Graphen dar. Die Verbindungslinien zwischen den Mitspielern, also alle möglichen Schusslinien, bilden die Kanten des Graphs. Startknoten ist der Spieler im Ballbesitz, der Zielknoten das gegnerische Tor (Abb. 4 rechts). In diesem Graphen muss ein möglichst kurzer Weg zum gegnerischen Tor gefunden werden, bei dem eine möglichst geringe

Wahrscheinlichkeit besteht, den Ball an den Gegner zu verlieren.

Zur Lösung der Aufgabe wird der Dijkstra-Algorithmus eingesetzt. Der Dijkstra-Algorithmus ist ein klassischer Algorithmus zur Wegeplanung und setzt einen kantengewichteten Graphen mit nichtnegativen Gewichten voraus [6].

Aus den einem Spieler bekannten Vektoren zu Mitspielern, Gegnern und Ball werden zunächst die Kanten eines Graphs ermittelt. Die Schusslinien zum gegnerischen Tor sind ebenfalls Bestandteil des Graphen.

Im nächsten Schritt müssen die Kantengewichte bestimmt werden. Hierbei wird die Länge der Kante berücksichtigt, da bei kürzeren Wegen ein präziseres Spielen möglich ist. Außerdem muss das Kantengewicht in Abhängigkeit von den Positionen der Gegner bestimmt werden, um zu verhindern, dass Pässe gespielt werden, bei denen der Ball von einem Gegner abgefangen werden kann. Eine Kante, die nahe an einem Gegenspieler vorbeiführt, bekommt ein hohes Kantengewicht. So wird es wenig wahrscheinlich, dass in diese Richtung gespielt wird.

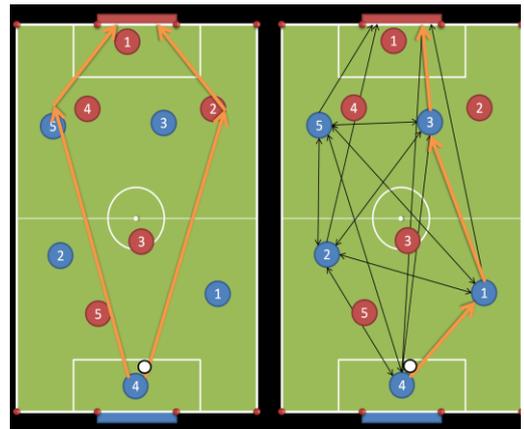
Da das Tor nicht ein punktförmiges Ziel ist, muss der optimale Torschusswinkel bestimmt werden. Ist das Tor leer, so ist es sinnvoll, in die Mitte des Tores zu schießen. Meist befindet sich aber ein gegnerischer Spieler im Tor, an dem vorbeigeschossen werden muss.

3.4 ERGEBNISSE

Abb. 4 zeigt den Nutzen einer solchen Planung.

Nach der einfacheren Logik (Abb. 4, links) hat der Torwart, der Spieler mit der Nummer 4, im Ballbesitz nur die Möglichkeiten, auf die linke und die rechte Stürmerposition zu spielen. Spielt er seinen Mitspieler mit der Nummer 5 an, hat dieser keine gute Möglichkeit, den Ball ins Tor zu schießen, da der Weg durch einen Gegner blockiert ist.

Indem er einen Graphen mit sich selbst als Startknoten aufbaut, kann der Spieler im Ballbesitz einen Weg bis zum gegnerischen Tor vorausplanen und dabei die Positionen aller Mitspieler und Gegner berücksichtigen. Dadurch findet er einen besser spielbaren Weg. Im Beispiel spielt der Torwart auf Spieler 1, der Spieler 3 anspielen kann, der eine gute Schussmöglichkeit zum Tor hat (Abb. 4, rechts).



**Abbildung 4: Planung des Zuspies:
links ohne, rechts mit Verwendung des
Graphenalgorithmus**

Bei diesem Vorgehen plant jeder Agent einen Weg, der für die aktuell vorliegende Situation optimal ist. Wenn sich die Situation nicht verändert, wird der angespielte Spieler zur selben Beurteilung der Situation kommen und den begonnenen Plan fortführen. Ändert sich die Situation in der Zwischenzeit, bestimmt er einen neuen Plan und korrigiert die begonnene Strategie.

Je mehr Spieler in einer Mannschaft spielen, desto wichtiger wird es, eine Zuspiesstrategie zu entwickeln. Dies zeigte sich 2011 beim Umstieg von fünf auf sieben Spieler pro Mannschaft. Die RT Lions sind dank der hier vorgestellten Zuspiesstrategie den mit der einfacheren Strategie spielenden Mannschaften deutlich überlegen.

3.5 AUSBLICK

Von 2012 an soll in der Mixed-Reality-Liga als erster Liga mit 11 Spielern pro Mannschaft gespielt werden. Aufgrund des größer werdenden Graphen wird dann die verfügbare Rechner-Kapazität auf dem Client-Rechner einen begrenzenden Faktor darstellen. Deswegen sollten zukünftig Algorithmen wie der A*-Algorithmus eingesetzt werden, die zusätzlich eine Heuristik verwenden, um den Suchraum einzuschränken und so die Laufzeit zu verringern [7].

Im vorliegenden Ansatz wird davon ausgegangen, dass die Positionen der Spieler den aktuell gemessenen entsprechen. In der Realität haben Spieler die Möglichkeit, günstige Positionen im Graphen anzufahren. Zu untersuchen ist, wie sich die Theorie dahin gehend verwenden lässt, dass Spieler einen Graphen mit günstigen Kantengewichten aufbauen.

4. PÄDAGOGISCHER NUTZEN

Im Roboterfußball werden die Mechanismen in einem System reaktiver Softwareagenten spielerisch visualisiert. So lassen sich Studierende leicht an die Thematik heranführen.

Seit 2009 nimmt die Hochschule Reutlingen regelmäßig an Wettbewerben der Mixed Reality teil. Studentische Teams befassen sich dabei unter Anleitung von Assistenten mit der Weiterentwicklung von Hard- und Software. Im Rahmen des Projektes sind bereits mehrere Abschlussarbeiten entstanden [2,4,6,8]. Die Verbesserungen der Hardware kommen allen Mannschaften der Liga zu Gute und führen zu einer Zusammenarbeit studentischer Teams aller beteiligten Hochschulen. Entscheidend für das Abschneiden der eigenen Mannschaft bei Wettbewerben ist jedoch die Weiterentwicklung der künstlichen Intelligenz. Der durch die Wettbewerbe aufgebaute Konkurrenzgedanke treibt die Entwicklung voran. Neben den eigentlichen Entwicklungsaufgaben lernen die Studierenden auch, Entwicklungen unter Zeit- und Leistungsdruck durchzuführen, und ihre Fähigkeiten zur Organisation im Team und im Projekt werden geschult.

2009 gelang den RT Lions bei der Weltmeisterschaft in Graz auf Anhieb der Sprung an die Spitze der Liga. Die hier dargestellten Ergebnisse wurden erstmals zur Weltmeisterschaft in Singapur 2010 eingesetzt, wo es den RT Lions gelang, den 2. Platz in der Mixed-Reality-Liga zu belegen [6]. 2011 erreichte das Team bei den offenen iranischen Meisterschaften in einer internationalen Konkurrenz wieder den ersten Platz.

5. DANKSAGUNG

Das Projekt wird wesentlich durch Mittel aus der Industriepartnerschaft des Studienbereiches Mechatronik unterstützt [9].

6. LITERATURVERZEICHNIS

- [1] RoboCup Startseite. RoboCup Germany http://www.robocupgermanopen.de/de/_robocup. Letzter Zugriff 19.04.2011
- [2] Weber, M. 2010. Verbesserung der Positionsschätzung fahrbarer Miniroboter mittels Kalmanfilter für den Einsatz beim RoboCup. Bachelor-Thesis, Hochschule Reutlingen.
- [3] Wooldridge, M. 2002. *Intelligent Agents: The Key Concepts*. Springer Berlin / Heidelberg S. 151-190, doi:10.1007/3-540-45982-0_1
- [4] Kurz, A. 2009. Künstliche Intelligenz eines Multiagentensystems. Bachelor-Thesis, Hochschule Reutlingen.
- [5] Segewick, R. 2004. Algorithms in Java, 3rd edition, part 5: graph algorithms. Pearson Education
- [6] Bergler, Hannes. 2010. Verbesserung der Kooperation in Multiagentensystemen durch Anwendung von Graphenalgorithmen. Bachelor-Thesis, Hochschule Reutlingen.
- [7] Russell S., Norvig P. 2004. *Künstliche Intelligenz – Ein moderner Ansatz*, Prentice Hall, ISBN 3-8273-7089-2.
- [8] Ott, S. 2009. Aufbau und Inbetriebnahme einer RoboCup Mixed Reality Spiel- und Testumgebung. Bachelor-Thesis, Hochschule Reutlingen.
- [9] Industriepartnerschaft Mechatronik. Hochschule Reutlingen. <http://www.tec.reutlingen-university.de/mechatronik/mechatronik-studium/stipendien-industriepartnerschaft.html>. Letzter Zugriff 19.04.2011

Tierverhalten auf dem Roboter Asuro

Alexander Rometsch

Schülerlabor Neurowissenschaften für Integrative Neurowissenschaften
Hertie-Institut für klinische Hirnforschung

alexander.rometsch@student.reutlingen-university.de

ABSTRACT

Die Arbeit behandelt die Einführung von Schülern in den Bereich der Untersuchung von tierischem Verhalten und deren Abbildung auf den Roboter „Asuro“.

1. PROJEKT ASURO

Der Roboter Asuro (Another Small And Unique Robot from Oberpfaffenhausen) wurde vom Deutschen Zentrum für Luft- und Raumfahrt am Institut für Robotik und Mechatronik entwickelt. Der Asuro (Abbildung 1) besteht, bis auf die Platine, aus handelsüblichen Bauteilen. Er lässt sich in der Sprache C programmieren. Es existiert eine für den Asuro konzipierte Bibliothek, diese liefert viele Funktionen (z.B. geradeaus fahren, Abfrage der Sensorwerte), um den Einstieg in die Programmierung zu erleichtern.

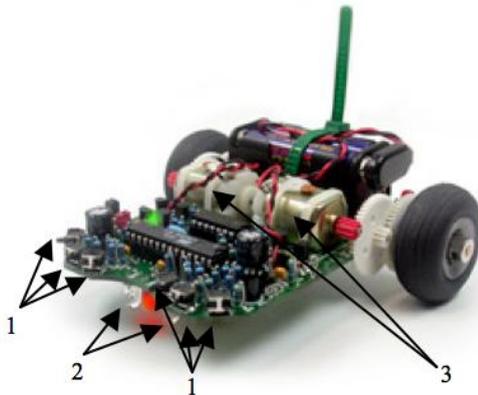


Abbildung 1: Asuro¹

Die Steuerung des Roboters übernimmt ein RISC-Prozessor. Die benötigte Energie liefern vier AAA Zellen. Für den Antrieb sorgen zwei Elektromotoren (3), welche sich unabhängig voneinander ansteuern lassen. An der Vorderseite besitzt der Asuro eine optische Linienfolgeeinheit (2) und sechs Kollisionstaster (1). Zudem befinden sich verschiedene LEDs auf der Platine, welche ein- bzw. ausgeschaltet werden können und eine Infrarot-

Kommunikationseinheit, über die Programmierung und Fernsteuerung möglich sind. [1][2]

2. UNTERSUCHUNG TIERISCHEN VERHALTENS

Die Schüler untersuchten das Verhalten verschiedener Tiere und bildeten dieses, soweit wie möglich, auf dem Asuro ab.

2.1 PANTOFFELTIERCHEN (PARAMECIUM)

Das Pantoffeltierchen (Abbildung 2) ist ein Einzeller und lebt im Süßwasser. Es kommt in Teichen, Seen, Flüssen und Pfützen vor. Unter dem Mikroskop wurde das Verhalten des Pantoffeltierchens untersucht. Zur Fortbewegung lässt sich zusammengefasst sagen, dass das Pantoffeltierchen sich ständig im Wasser hin und her bewegt. Sobald es auf einen Widerstand trifft, schwimmt es zurück und ändert seine Richtung zufällig.



Abbildung 2: Paramecium²

Die Abbildung auf den Asuro lässt sich mit Hilfe der Kollisionstaster erreichen. Der Asuro fährt so lange geradeaus, bis die Kollisionstaster einen Widerstand melden. Darauf bewegt sich der Asuro ein Stück zurück, dreht sich zufällig und fährt weiter. [3]

¹http://www.arexx.com/data/images/asuro/asuro_front.jpg Letzter Zugriff: 14.03.2011

²<http://www.scienceblogs.de/astrodicticum-simplex/2009/12/01/Paramecium.jpg> Letzter Zugriff: 14.03.2011

2.2 BILDUNG EINER AMEISEN-STRABE

Bei der Futtersuche laufen Ameisen kreuz und quer, bis sie eine Nahrungsquelle gefunden haben. Auf dem Rückweg legt jede Ameise eine Duftspur (Pheromonspur) zum Bau. Durch die Duftspur finden weitere Ameisen aus dem Bau mit Hilfe ihrer Fühler den Weg zur Nahrungsquelle. Um das Verhalten bei der Futtersuche auf den Asuro abzubilden, wird auf die optische Linienfolgeinheit zurückgegriffen. Eine schwarze Linie dient als Pheromonspur (Abbildung 3). Der Asuro besitzt zwei Lichtsensoren mit der die Lichtintensität gemessen werden kann. Die Lichtintensität auf weißem Untergrund ist höher, als die auf dunklem Boden. Mit den unabhängig voneinander ansteuerbaren Elektromotoren, kann durch gezielte Verringerung oder Erhöhung der Geschwindigkeit eines Motors in Abhängigkeit der Lichtintensität, der Asuro der schwarzen Linie folgen. [4][5]



Abbildung 3: Ameisenverhalten

3. SUMO-RINGER

Neben Tierverhalten wurde auch der Kampf von Sumo-Ringern mit dem Asuro simuliert.

3.1 SUMO-RINGER

Beim Sumo-Ringen versucht der Kämpfer seinen Gegner zu Fall zu bringen oder aus dem Kreis zu stoßen. Wegen der Bauform des Asuros, ist es nicht möglich, so einen weiteren Asuro zu Fall zu bekommen. Deshalb kann ein Asuro, einen weiteren Asuro nur aus dem Kreis schieben. Zur Vereinfachung wurde statt einem Kreis ein Viereck, als Arena gewählt. Für die Umsetzung werden Auswertungen der Lichtsensoren (um im Feld zu bleiben) und der Kollisionstaster (Erkennung des Gegners) benötigt.

Die zwei Asuros fahren immer geradeaus. Fall1: Asuro erreicht die Markierung des Kampffeldes. Zurückfahren, drehen und weiter fahren. Fall2: Asuro berührt mit den Kollisionstastern den anderen Asuro. Hintere rote LEDs einschalten, beschleunigen und anderen Asuro aus dem Kampffeld schieben. Sobald die Markierung des Feldes erreicht wird tritt Fall1 ein.

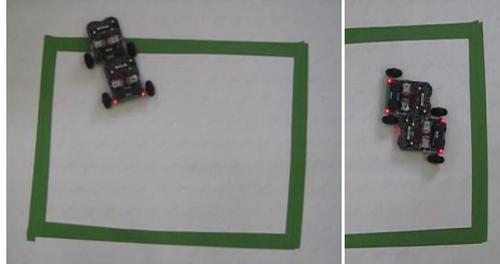


Abbildung 4 und 5: Sumo-Ringen mit zwei Asuros

Durch die Bauform des Asuros kann es vorkommen, dass sich die beiden Asuros ineinander verhaften und nicht mehr aus der Situation heraus kommen („Sumo-Tanz“).

4. LITERATURVERZEICHNIS

- [1] Arexx Website. <http://www.arexx.com/>, Letzter Zugriff: 14.03.2011
- [2] DLR Website. <http://www.dlr.de/>, Letzter Zugriff: 14.03.2011
- [3] Fritz Bramstedt, Dressurversuche mit *Paramecium caudatum* und *Stylonychia*, Volume 22, Number 4, 490-516, DOI: 10.1007/BF00572723
- [4] Nina Marwede, 2007, Entwicklung von Ameisenstraßen mit virtuellen Pheromonen. <http://ninamarwede.de/pub/Marwede2007EntwicklungVonAmeisenstrassenMitVirtuellenPheromonen.pdf>. Universität Oldenburg. Letzter Zugriff: 14.03.2011
- [5] Dr. Ute Vogel, Von der Natur lernen- Optimierungsverfahren der Informatik, <http://www-ui.informatik.uni-oldenburg.de/download/RintelnVL090727.pdf> Universität Oldenburg, Letzter Zugriff: 14.03.2011

Anwendungsentwicklung zur Untersuchung motorischen Lernens

Alexander Rometsch

Schülerlabor Neurowissenschaften für Integrative Neurowissenschaften
Hertie-Institut für klinische Hirnforschung

alexander.rometsch@student.reutlingen-university.de

ABSTRACT

Diese Arbeit befasst sich mit dem Thema motorisches Lernen beim Menschen. Zur Untersuchung des motorischen Lernens wurde eine Anwendung entwickelt, welche im Zuge einer beginnenden Diplomarbeit neue Kenntnisse gewinnen soll.

1. ANWENDUNGS-ENTWICKLUNG MOTORISCHES LERNENS

Zur Untersuchung des motorischen Lernens beim Menschen soll eine Anwendung entwickelt werden, mit der Messungen aufgezeichnet werden. Mit Hilfe der Messergebnisse verschiedener Testprobanden, soll die Lernfähigkeit untersucht werden.

1.1 LERNEN

Definition Lernen nach Schmidt 1975 (zitiert nach [2]):

„Lernen ist eine relativ andauernde Veränderung der Fähigkeit zu antworten, die aus der Übung oder Erfahrung hervorgeht.“

1.2 MOTORISCHES LERNEN ALS PROZESS

Lernen ist ein aktiver Prozess für Körper und Gehirn. Übungen, die den Lernenden einbeziehen fördern das Lernen. Dabei sind vorgefertigte Bewegungen nur wenig hilfreich. Beim motorischen Lernprozess werden ankommende Informationen wahrgenommen und im sensomotorischen Gedächtnis dekodiert. Die Informationen werden mit bereits vorhandenen Erfahrungen, Kenntnissen und Fertigkeiten abgeglichen. Danach handelt der Mensch und führt eine Aktion aus. Diese Aktion wird mit dem ursprünglichen Ziel verglichen und bei Abweichungen werden Korrekturen eingeleitet. Beim Lernen bilden sich Synapsen, welche wiederum mit den Nervenzellen verbunden sind. Je häufiger bestimmte Verbindungen benutzt werden, desto besser ist die Impulsübertragung und somit die automatische Bewegungsausführung. Beim Lernen von motorischen Mustern werden diese als implizite Leistung im prozeduralen Gedächtnis abgespeichert. Durch Erfolge wird die Motivation beeinflusst und Fehler fördern das Lernen. [1][2]

2. LERNPHASEN

Nach verschiedenen Autoren lassen sich 2 bis 5 Lernstadien beschreiben. Generell wird jedoch von 3 Phasen gesprochen.

Autonome oder automatische Phase:

- Die Informationsaufnahme läuft vollständig automatisch ab. Der Lernende bekommt davon nichts mit. Lernende können in diesem Stadium nicht mehr durch externe Reize gestört werden. Oft können zwei Dinge zur selben Zeit ausgeführt werden.
- Das Zentralnervensystem ist an die Umgebungsbedingungen angepasst. Die Fähigkeit vorherzusagen, ob ein Ereignis eintritt ist vollständig entwickelt (Antizipation).
- Beobachtete Bewegungen werden automatisch ausgeführt. Für die Aufnahme der Informationen einer neuen Bewegungs-ausführung wird nur ein geringer Anteil an Aufmerksamkeit benötigt. Entstehen Fehler in der Ausführung, werden diese selbstständig entdeckt und korrigiert. Neue Fortschritte sind kaum wahrnehmbar.

Assoziative Phase:

- Lernende nehmen zunehmend Informationen automatisch auf, benötigen aber immer noch relativ viel Aufmerksamkeit. Externe Reize gelten in diesem Stadium als störend.
- Das Zentralnervensystem vergleicht neue Informationen mit vorhandenen und entwickelt Verknüpfungsmechanismen. Hauptaufgabe ist das Gruppieren und Differenzieren. Kriterien für den korrekten Vergleich entwickeln sich (Fehlerentdeckungsmechanismen). Dadurch können Fehler in Bewegungsabläufen entdeckt werden.
- Beobachtete Bewegungen werden feiner und flüssiger nachgeahmt. Fehler treten immer weniger auf.

Kognitive Phase:

- Lernende nehmen Informationen auf, sind stark durch externe Reize stöbar.

- Das Ziel des Zentralnervensystems, ist die Aufnahme von Informationen, um diese in Aufgaben zu wandeln. Daraus werden Bewegungsstrategien und –muster entwickelt. Dieses Stadium ist wegen der Informationsverarbeitung sehr zeitaufwändig und verlangt viel Aufmerksamkeit vom Lernenden.
- Beim Nachahmen des beobachteten Bewegungsverhaltens kommt es zu vielen Fehlern und der zeitliche Ablauf ist falsch. Die Bewegungsausführung variiert sehr stark, jedoch sind schnelle Fortschritte sichtbar. [1][2]

3. VERSUCHSAUFBAU

Für die Untersuchung müssen zuerst verschiedene Parameter festgelegt werden. Dadurch können dann in der Lern- bzw. Testphase verschiedene Szenarien durchlaufen werden. Zu den Parametern gehören: Anzahl der Lern- bzw. Testversuche, Geschwindigkeit des Mauszeigers, ein- bzw. ausschalten der Mauszeigerbeschleunigung.

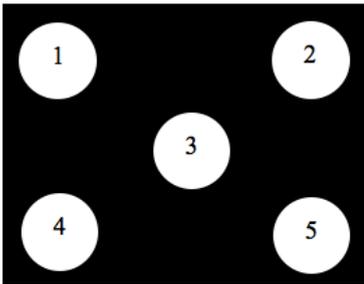


Abbildung 1: Möglichkeiten des Erscheinens

Bei jedem Versuch erscheint per Zufall auf dem Monitor ein kleines weißes Kästchen auf schwarzem Hintergrund an einem der fünf Erscheinungsmöglichkeiten (Abbildung 1). Dieses muss die Testperson mit dem Mauszeiger anklicken. Die Ausgangssituation des Mauszeigers ist die Bildschirmmitte. Das Kästchen kann an maximal fünf verschiedenen Stellen auftreten. Das Auftreten des Kästchens basiert auf einer Zufallsfunktion. Je nach Anzahl der Versuche wird ein Feld mit derselben Größe angelegt. Dieses Feld wird mit den Möglichkeiten des Erscheinens gefüllt. Bei 10 Versuchen sieht das Feld so aus: Feld [1,2,3,4,5,1,2,3,4,5]. Durch die Zufallsfunktion wird aus dem Feld zufällig ein Wert pro Versuch für die Erscheinung gewählt. Eine Feldposition wird nur einmal ausgewählt, damit jede Erscheinungsmöglichkeit auftreten kann.

3.1 VERSUCHSPHASE

Der Versuchsablauf ist in zwei Teile gegliedert, die Lernphase und die Testphase. Bei den Durchläufen

der Lernphase erscheint das weiße Kästchen an einer der fünf Positionen auf schwarzem Hintergrund. Kurze Zeit später wird der Mauszeiger auf der Bildschirmmitte sichtbar. Die Versuchsperson muss den Mauszeiger auf das Kästchen führen und mit linker Maustaste anklicken. Das Kästchen und der Mauszeiger verschwinden sofort. Es wird eine neue Position des Kästchens per Zufall ausgerechnet und das Kästchen erscheint an der neuen Position. Nun muss die Testperson wieder den Mauszeiger auf das Kästchen führen und mit linker Maustaste bestätigen. Die Wiederholung hängt von dem zu Beginn gesetzten Parameter der Versuchsanzahl ab. Während der Lernphase eignet sich die Versuchsperson den Umgang mit der Mausgeschwindigkeit an. Direkt nach der Lernphase startet die Testphase. Diese läuft identisch zu der Lernphase ab, unterscheidet sich aber in der Sichtbarkeit des Kästchens und des Mauszeigers. Bei der Lernphase sind Kästchen und Mauszeiger dauerhaft sichtbar. In der Testphase erscheint das Kästchen wenige Millisekunden, danach verschwindet es. Der Mauszeiger erscheint ebenfalls nur kurz auf der Bildschirmmitte und wird danach unsichtbar. Die Testperson muss versuchen den unsichtbaren Mauszeiger so nah wie möglich an die Stelle, an der das Kästchen erschienen ist, zu führen. Nachdem die Testphase abgeschlossen ist, können je nach Parametereinstellungen weitere Lern- und Testphasen mit anderen Mauszeigereinstellungen folgen. Die Aktionen der Versuchsperson werden abgespeichert. Dabei werden die Position des Kästchens und die Strecke des Mauszeigers dokumentiert. Anhand dieser Daten werden mehrere Diagramme gezeichnet. In den Diagrammen werden die x und y Werte des weißen Kästchens mit denen des Mauszeigers an der Stelle der Bestätigung durch die linke Maustaste verglichen. Weiterhin wird der zurück gelegte Weg des Mauszeigers in einer anderen Grafik dargestellt. Für zukünftige Messdaten sollen die Testpersonen mit Prismenbrillen (Umkehrbrillen) ausgestattet werden.

3.2 AUSBLICK

Die Anwendung zur Untersuchung des motorischen Lernens dient als Grundlage für eine demnächst beginnende Diplomarbeit. Im Laufe der genannten Testversuche wird sie den neuen Erkenntnissen angepasst.

3.3 WERKZEUGE

Die Anwendung zur Untersuchung des motorischen Lernens ist in MATLAB mit der Erweiterung Psychtoolbox entwickelt. MATLAB unterstützt die Einbindung von C/C++ Quellcode. Um den Mauszeiger zu manipulieren wurde ein Programm in C++ entwickelt, das über die Windows API die Einstellungen der Maus ändert.

4. LITERATURVERZEICHNIS

- [1] Eva Hendrich, Motorisches Lernen und kontralateraler Transfer, 2003, Universität München, http://edoc.ub.uni-muenchen.de/1563/1/Hendrich_Eva.pdf Letzter Zugriff: 14.03.2011
- [2] Barbara Suppé und Susanne Klein-Vogelbach, Irene Spirgi-Gantert und Barbara Suppé, FBL Klein-Vogelbach Functional Kinetics: Die Grundlagen Bewegungsanalyse, Untersuchung, Behandlung, Springer Medizin Verlag 2007

KÜHLUNG BEI LED-PROJEKTOREN: EIN VERGLEICH VON ZWEI KÜHLSYSTEMEN

Daniel Grbavac
Hochschule Reutlingen

daniel.grbavac@student.reutlingen-university.de

ABSTRACT

Die vorliegende Arbeit befasst sich mit verschiedenen Möglichkeiten zur Kühlung von LED-Projektoren. Dabei wird besonders auf zwei Kühlsysteme eingegangen: ein Heatpipe- und ein Wasserkühlsystem. Neben deren theoretischen Erläuterung erfolgt ein praktischer Vergleich, wobei in mehreren Versuchen die Kühlfähigkeit der beiden Systeme ermittelt und die Ergebnisse anschließend gegenübergestellt werden.

1. EINLEITUNG

LEDs finden heutzutage in vielen Bereichen Anwendung – unter anderem bei Taschenlampen oder Ampeln. Auch bei Projektoren werden LEDs immer häufiger als Lichtquelle verwendet. Jedoch ergibt sich bei der Verwendung von LEDs das Problem, dass sie sich im Betrieb erwärmen. Um die LEDs eines Projektors vor einer Beschädigung durch Überhitzung zu schützen, muss folglich für eine ausreichende Kühlung gesorgt werden.

2. LED GRUNDLAGEN

LED steht für „Light Emitting Diode“ und bedeutet ins Deutsche übersetzt lichtemittierende Diode oder Leuchtdiode. Bei LEDs handelt es sich um Halbleiterdioden, die elektrische Energie in elektromagnetische Strahlung umwandeln. Dies geschieht gewöhnlich in Form von sichtbarem Licht.

2.1 FUNKTIONSWEISE

Eine Leuchtdiode besteht aus zwei verschiedenen Schichten: einer n- und einer p-Schicht (siehe Abbildung 1). Die n-Schicht enthält Elektronen und die p-Schicht sogenannte Defektelektronen (Löcher), die wie positive Ladungsträger wirken. Zwischen den beiden Schichten befindet sich eine Grenzschicht, die als Raumladungszone bezeichnet wird [1].

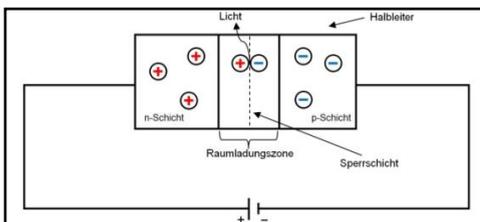


Abb. 1: Darstellung einer Halbleiterdiode [1]

Wird eine Spannung so angelegt, dass der Pluspol mit der p-Schicht und der Minuspol mit der n-Schicht verbunden ist, so wirkt die LED als Leiter. Dadurch bewegen sich die Elektronen von der n-zur p-Schicht und die Löcher genau umgekehrt. Indem sich die Elektronen und Löcher im Bereich der Sperrschicht verbinden, wird Licht ausgesendet.

2.2 THERMISCHE ASPEKTE

Der Wirkungsgrad einer Glühlampe liegt bei etwa 5 Prozent [2]. LEDs arbeiten im Vergleich dazu zwar effizienter, haben aber dennoch das Problem, dass sie einen Großteil der elektrischen Energie in Wärme umwandeln und dadurch heiß werden. Um die Leuchtdioden vor Überhitzung zu schützen, muss daher für ausreichend Kühlung gesorgt werden, die z.B. durch die Verwendung von Kühlkörpern erreicht werden kann.

Der elektrische Strom ist ein Faktor, der die Temperatur einer LED beeinflusst. Dieser fließt bei einer LED erst ausreichend, wenn eine sogenannte Schwellspannung erreicht ist. Eine anschließend geringfügige Spannungserhöhung hat zur Folge, dass die Stromstärke exponentiell steigt, wodurch sich die Leuchtdiode stärker erwärmt und sogar beschädigt werden kann.

Die Betriebstemperatur ist neben dem elektrischen Strom ein weiterer und wichtiger Aspekt, der die Lebensdauer von Leuchtdioden beeinflusst. Im Allgemeinen gilt: je kühler die LED, desto effizienter ist sie. Höhere Temperaturen führen dagegen zu einer Verringerung der Lichtausbeute und verkürzen die Lebensdauer. Durch zahlreiche Versuche am Lighting Research Center des Rensselaer Polytechnic Institute wurde gezeigt, dass die Lebensdauer der LEDs mit zunehmender Temperatur exponentiell sinkt. Eine Temperaturerhöhung von zehn Grad kann unter anderem dazu führen, dass die Lebensdauer einer Leuchtdiode halbiert wird [3][4].

Die Ausführungen machen deutlich, dass die Kühlung von Leuchtdioden nicht vernachlässigt werden darf, um eine optimale Lichtausbeute sowie eine lange Lebensdauer zu ermöglichen.

3. KÜHLSYSTEME

Um die LEDs eines Projektors vor Überhitzung zu schützen, gibt es verschiedene Lösungen: Eine davon sind Kühlkörper und eine andere sogenannte

Heatpipes; aber auch der Einsatz von Kühlflüssigkeiten, wie zum Beispiel Wasser, ist möglich. Im folgenden Kapitel wird erläutert, wie die genannten Kühlarten funktionieren.

3.1 KÜHLKÖRPER

Ein Kühlkörper besteht aus einem wärmeleitenden Material (Kupfer oder Aluminium). Er sorgt für eine verbesserte Wärmeabführung an einem wärmeerzeugenden Bauteil, indem er dessen Oberfläche vergrößert. Dadurch wird das Bauteil vor einem Ausfall oder einer Beschädigung durch Überhitzung geschützt. Diese Art von Kühlung wird als passive Kühlung bezeichnet. Durch die Verwendung eines Lüfters kann zusätzliche Luft entlang des Kühlkörpers geführt werden, sodass eine höhere Kühlleistung erreicht wird. In diesem Fall spricht man von aktiver Kühlung.

3.2 HEATPIPES

Heatpipes (zu Deutsch: Wärmerohre) sind spezielle Rohre, die Wärme effizient transportieren können. Dazu nehmen sie die an einem Ort anfallende Wärme möglichst schnell auf und leiten diese an einen anderen Ort weiter, an dem eine Abkühlung ermöglicht wird.

3.2.1 AUFBAU & FUNKTIONSWEISE

Eine Heatpipe besteht aus einem hochleitenden Material (zumeist aus Kupfer) und einer Flüssigkeit, die sich im Inneren des Rohres befindet.

Abbildung 2 veranschaulicht die Funktionsweise von Heatpipes.

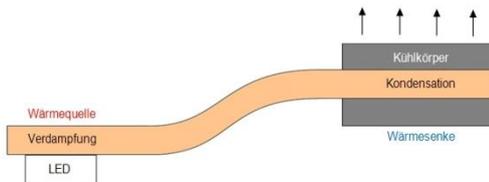


Abb. 2: Funktionsprinzip Heatpipe-Kühlsystem

Die Heatpipe hat an einem Ende Kontakt zu einer Wärmequelle, wie zum Beispiel einer LED. Dadurch kann die Heatpipe die von der Wärmequelle entstehende Energie aufnehmen. Es findet also ein Wärmeeintrag statt. Dieser hat zur Folge, dass sich die Kühlflüssigkeit, die auch als Arbeitsmittel bezeichnet wird, erwärmt und schließlich durch den in der Heatpipe herrschenden Unterdruck verdampft. Der entstandene Dampf bewegt sich nun mit der enthaltenen Wärmeenergie von der Wärmequelle zum anderen Ende der Heatpipe, das als Wärmesenke bezeichnet wird. Hier ist die Heatpipe mit einem Kühlkörper verbunden. So kann der Dampf seine Wärmeenergie an diesen abgeben und kondensieren. Das Kondensat fließt daraufhin

zurück zur Wärmequelle, wo es wieder erwärmt wird und sich der Vorgang wiederholt. [5]

3.3 WASSERKÜHLUNG

Unter Wasserkühlung, die auch als Flüssigkeitskühlung bezeichnet wird, versteht man ein Kühlsystem, bei welchem die Wärmeenergie der wärmeerzeugenden Komponenten mittels Wasser abgeführt wird.

Bei den im Rahmen dieser Arbeit durchgeführten Untersuchungen (siehe Kapitel 4) wurde ein Wasserkühlsystem verwendet, das dem einer PC-Wasserkühlung entspricht.

3.3.1 KOMPONENTEN

Die **Pumpe** ist ein wichtiges Bauteil einer Wasserkühlung. Sie befördert das Wasser durch die einzelnen Bestandteile des Kühlsystems.

Die **Kühlkörper** eines wassergekühlten Systems bestehen aus einem gut wärmeleitenden Material. Sie werden auf das zu kühlende Element, wie zum Beispiel eine LED montiert, sodass sie die dort entstehende Wärme in den Wasserkreislauf abführen. Dafür haben die Kühlkörper entsprechende Schlauchanschlüsse, damit das Wasser durch sie hindurchfließen kann.

Radiatoren sind Wärmeüberträger und bestehen meistens aus vielen dünnen Lamellen oder Kupferrohren, durch die das Wasser fließt. Sie sind für die Kühlung des im System vorhandenen Wassers zuständig. Aktive Radiatoren besitzen zusätzliche Lüfter, durch welche eine effizientere Kühlung des durchfließenden Wassers ermöglicht wird als dies bei passiven Radiatoren der Fall ist, die ohne Lüfter arbeiten.

In einem **Ausgleichsbehälter (AGB)** wird das Wasser des Kühlsystems gesammelt. Er dient zum Befüllen und Entlüften des Wasserkreislaufs und bietet der Kühlflüssigkeit bei erwärmungsbedingter Ausdehnung genügend Platz.

Um die einzelnen Komponenten miteinander zu verbinden sind Schläuche und verschiedene Anschlüsse notwendig.

3.3.2 FUNKTIONSWEISE

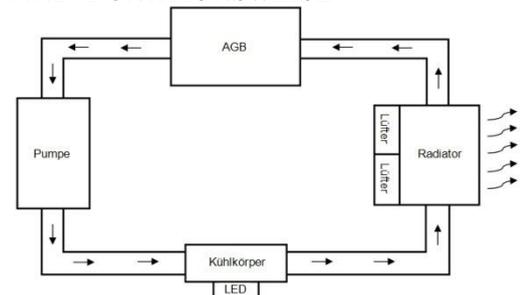


Abb. 3: Darstellung einer Wasserkühlung

Die Pumpe befördert das Wasser durch die einzelnen Komponenten im Kühlkreislauf. Dabei nimmt das Wasser die Wärme der zu kühlenden Elemente am Kühlkörper auf und transportiert sie anschließend zum Radiator. Hier wird das Wasser letztendlich gekühlt, indem die zuvor aufgenommene Wärmeenergie bzw. ein Teil davon an die Umgebungsluft abgegeben wird.

4. VERGLEICH

Bei LED-Projektoren können die in Kapitel 3 angeführten Kühlmöglichkeiten zur Kühlung von LEDs verwendet werden. Da die Projektoren jedoch mit Hochleistungs-LEDs betrieben werden, ist es nicht möglich, gewöhnliche, passive Kühlkörper zur Kühlung dieser Leuchtdioden zu verwenden. Ihre Kühlleistung ist dafür zu gering. Für die vorhandenen Hochleistungs-LEDs kommt somit die Kühlung mittels eines Heatpipe-Systems oder einer Flüssigkeitskühlung in Frage. Um die Kühleffektivität dieser beiden Kühlsysteme beim LED-Projektor aufzuzeigen und zu vergleichen, wurden verschiedene Versuche durchgeführt. Die dabei eingesetzten Projektoren waren außer der Kühlung identisch: die Leuchtdioden wurden bei einem Projektor durch Heatpipes und beim anderen mit Hilfe von Wasser gekühlt.

4.1 VORBEREITUNG

4.1.1 AUFBAU DER KÜHLSYSTEME

Abbildung 4 zeigt exemplarisch den Aufbau des Kühlsystems mit den Heatpipes.

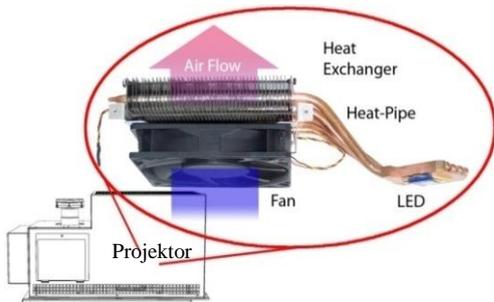


Abb. 4: Aufbau der Heatpipe-Kühlung [6]

Wie zu sehen ist, transportieren mehrere Heatpipes die Wärme von der Leuchtdiode zum entsprechenden Kühlkörper, der von einem Lüfter gekühlt wird. Die dabei verwendeten Heatpipes bestehen aus Kupfer und die Kühlkörper, welche die Wärme an die Umgebung abgeben, aus Aluminium.

Das Wasserkühlsystem des LED-Projektors wurde mit den beschriebenen Komponenten (siehe Abschnitt 3.3.1) aufgebaut. Der Aufbau des Wasserkühlsystems entsprach größtenteils dem aus Abbildung 3. Der Unterschied liegt darin, dass nicht nur eine LED, sondern drei parallel gekühlt wurden.

Die verwendeten Kühlkörper bestehen aus Kupfer und der Radiator des Wasserkühlsystems sorgt mit zwei Lüftern für die Kühlung des Wassers.

4.1.2 BILDSIGNAL

Zur Durchführung der Messungen wurden die LED-Projektoren über ein DVI (Digital Visual Interface) Kabel mit einem HD-Abspielgerät verbunden. Die Ausgabe des Abspielgeräts wurde auf die native Auflösung der Projektoren eingestellt. Sie beträgt 1920x1080 Pixel bei einer Frequenz von 60 Hertz.

4.1.3 LEUCHTDIODEN

Bei den verwendeten Leuchtdioden (Rot, Grün und Blau) handelt es sich um Hochleistungs-LEDs der Firma Luminus des Typs PhlatLight PT120. Wie aus Kapitel 2 bekannt, produzieren LEDs im Betrieb Wärme. Diese Wärme führt zu einer höheren LED-Temperatur, die bei den PhlatLight PT120 an einem Sensor bzw. dem sogenannten Thermistor als T_{ref} ausgelesen werden kann (siehe Abbildung 5).

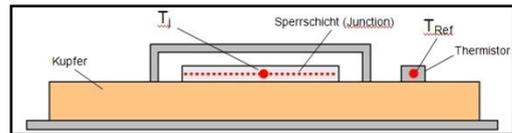


Abb. 5: Seitliche Ansicht vom LED Chipsatz [7]

Die PT 120 LEDs können eine Lebensdauer von über 60 000 Stunden erreichen, wenn folgende Temperaturen an der Sperrschicht T_j nicht überschritten werden: [7]

Tabelle 1: Maximal zulässige Temperaturen

Rote LED	Grüne LED	Blaue LED
$T_j < 80 \text{ }^\circ\text{C}$	$T_j < 125 \text{ }^\circ\text{C}$	$T_j < 115 \text{ }^\circ\text{C}$

Allerdings lässt sich die Sperrschichttemperatur bei der PT 120 nicht direkt messen. Sie muss mit der am Thermistor gemessenen Temperatur T_{ref} , der Verlustleistung der LED und mit dem Wärmewiderstands nach folgender Formel berechnet werden:

$$T_j = T_{ref} + (R_{(\delta-ref)} * P_{verlust}) \quad (4.1)$$

4.2 MESSUNGEN

Während der Projektor in Betrieb ist, kann die Temperatur einer Leuchtdiode über den Sensor und einer Computersoftware ausgelesen werden. Die gemessene Temperatur wird als T_{ref} bezeichnet und muss von der Sperrschichttemperatur T_j unterschieden werden (siehe Abschnitt 4.1.3).

4.2.1 AN- UND ABKLINGPHASE

Für einen ersten Vergleich der beiden Kühlsysteme wurden die drei Leuchtdioden nach genau einer Minute eingeschaltet und über einen Zeitraum von 10 Minuten betrieben. Die LED-Intensität betrug dabei 100%. Dieser Versuch wurde durchgeführt,

um den Temperaturanstieg nach dem Einschalten der LEDs zu beurteilen.

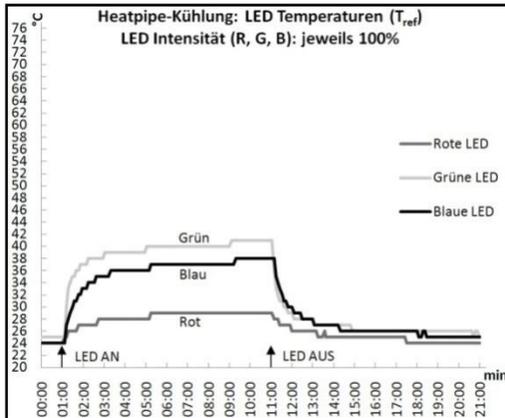


Abb. 6: Temperaturverlauf Heatpipe-Kühlung

Wie in Abbildung 6 zu erkennen ist, steigen die Temperaturen nach dem Einschalten der Leuchtdioden direkt und deutlich an. Der Grund dafür ist, dass die Leuchtdioden Wärmeenergie erzeugen, die nicht vollständig von den Heatpipes aufgenommen und an die Kühlkörper abgegeben werden kann. Denn eine gute Wärmeabführung kann erst dann erfolgen, wenn eine gewisse Temperaturdifferenz zwischen Wärmequelle und -senke bei den Heatpipes erreicht wird. Nach dem anfänglich deutlichen Temperaturanstieg treten beim Heatpipe-Kühlsystem ab ca. 5 bis 6 Minuten keine großen Temperaturänderungen an den LEDs mehr auf. Bei einer Umgebungstemperatur von ca. 23 °C, einer Betriebsdauer von 10 Minuten und einer Intensität von 100% an allen Leuchtdioden wurden folgende Maximaltemperaturen gemessen:

Tabelle 2: Maximal gemessene Temperaturen

Heatpipe-kühlung	Rote LED	Grüne LED	Blaue LED
	29 °C	41 °C	37 °C

Nach dem Abschalten der Leuchtdioden (bei 11 Minuten), fallen die Temperaturen anfangs sehr deutlich. Nach einer Abklingzeit von 10 Minuten erreichen die LEDs beinahe ihre Ausgangstemperaturen wieder (21. Minute).

Nachdem die LED-Temperaturen beim Heatpipe-Kühlsystem ermittelt wurden, wurde unter den gleichen Bedingungen die Wasserkühlung getestet. Auch hier findet ein schneller Anstieg der LED-Temperaturen statt, nachdem sie bei Minute eins angeschaltet wurden (siehe Abbildung 7). Die maximal gemessenen Temperaturen der Leuchtdioden bei der Wasserkühlung sind jedoch um einiges höher als beim System, welches durch Heatpipes gekühlt wird. Insbesondere bei der grünen LED sind die Temperaturunterschiede deutlich zu sehen. Während beim Heatpipe-Kühlsystem für die grüne

LED eine Temperatur von 41 °C gemessen wurde, erreichte sie beim wassergekühlten System 74 °C.

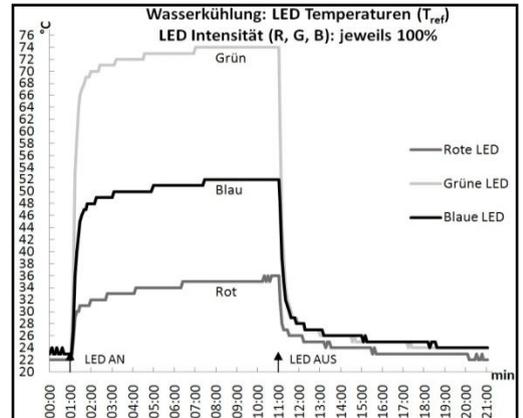


Abb. 7: Temperaturverlauf Wasserkühlung

Tabelle 3: Maximal gemessene Temperaturen

Wasser-kühlung	Rote LED	Grüne LED	Blaue LED
	36 °C	74 °C	52 °C

Ähnlich wie beim Heatpipe-Kühlsystem fallen bei der Wasserkühlung die Temperaturen der Leuchtdioden sehr deutlich, nachdem diese abgeschaltet wurden (bei 11 Minuten). Auch hier erreichen die LEDs bei Minute 21 nahezu ihre Ausgangstemperaturen.

Vergleich der Maximaltemperaturen

Um zu bestimmen, ob die Temperaturen der Leuchtdioden bei diesem Versuch unterhalb der kritischen T_j Werte des LED-Herstellers geblieben sind (siehe Abschnitt 4.1.3), mussten die jeweiligen Sperrschichttemperaturen berechnet werden. Diese Temperaturen sind in Abbildung 8 dargestellt.

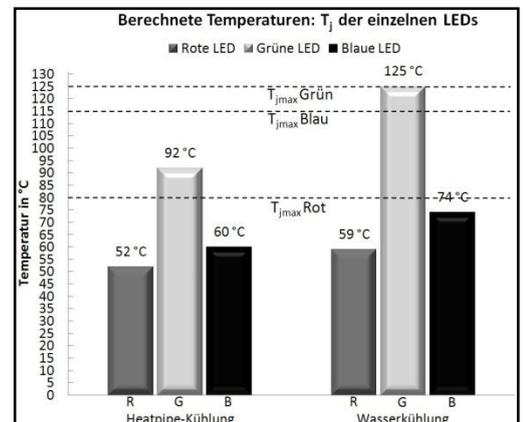


Abb. 8: Vergleich der T_j Temperaturen

Wie in Abbildung 8 zu erkennen ist, bleiben beim Heatpipe-Kühlsystem die T_j Temperaturen der drei Leuchtdioden deutlich unterhalb der Grenzwerte,

sodass eine lange Betriebsdauer möglich ist. Bei der Wasserkühlung sind die T_j Temperaturen der LEDs insgesamt höher als die der Heatpipe-Kühlung, wobei die rote und blaue Leuchtdiode immer noch ausreichend gekühlt werden. Nur bei der grünen LED könnte es auf Dauer zu Problemen kommen, da bei ihr eine Sperrschichttemperatur von 125 °C erreicht wurde. Diese entspricht exakt dem Grenzwert, der nicht überschritten werden darf, um eine Lebensdauer von über 60 000 Stunden zu erreichen.

4.2.2 VARIABLE UMGEBUNGSTEMPERATUR

Um herauszufinden, wie gut die Kühlsysteme die Verlustwärme der LEDs bei unterschiedlichen Umgebungstemperaturen abführen, wurden die LED-Projektoren bei Umgebungstemperaturen von 4, 8, 17, 23, 28 und 42 °C über einen Zeitraum von jeweils 10 min und einer Intensität von 100% an allen drei Leuchtdioden betrieben. Bei den genannten Temperaturen handelt es sich um die Sperrschichttemperaturen der einzelnen LEDs, die mit Formel (4.1) berechnet wurden.

Die Abbildungen 9 und 10 zeigen, welche Temperaturen nach 10 Minuten bei verschiedenen Umgebungstemperaturen an den Leuchtdioden erreicht wurden. Beim Vergleich der beiden Schaubilder ist zu erkennen, dass die LED-Temperaturen des Heatpipe-Kühlsystems bei den jeweiligen Umgebungstemperaturen niedriger als die der Wasserkühlung sind.

Anhand der Geraden in den Abbildungen 9 und 10 wird erkennbar, dass es bei beiden Kühlsystemen einen linearen Zusammenhang zwischen Leuchtdioden- und Umgebungstemperatur gibt.

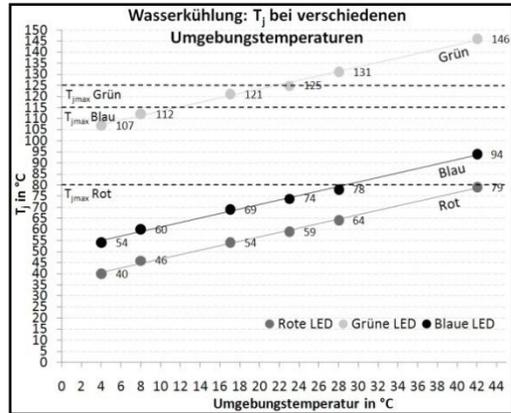


Abb. 10: T_j der LEDs, Wasserkühlung

4.2.3 AUSFALL VON KOMPONENTEN

Wie gut die Kühlung beim Ausfall wichtiger Komponenten funktioniert, sollte durch weitere Versuche untersucht werden. Dazu wurden die Lüfter der Kühlkörper beim Heatpipe-Kühlsystem sowie die Pumpe beim Wasserkühlsystem abgeschaltet, weshalb bei den folgenden Messungen höhere LED-Temperaturen als bei den bisherigen erwartet wurden. Um eine mögliche Beschädigung der Leuchtdioden zu vermeiden, wurden sie nicht mit 100%, sondern mit 40% betrieben. Außerdem wurde die Messdauer von 21 auf 51 min erhöht. Der Grund dafür waren die LED-Temperaturen, welche im Vergleich zu den bisherigen Messungen einerseits auch noch nach 10 oder 20 min beachtlich anstiegen und andererseits nach dem Abschalten der LEDs langsamer abfielen. Abbildung 11 und 12 veranschaulichen diese Messergebnisse.

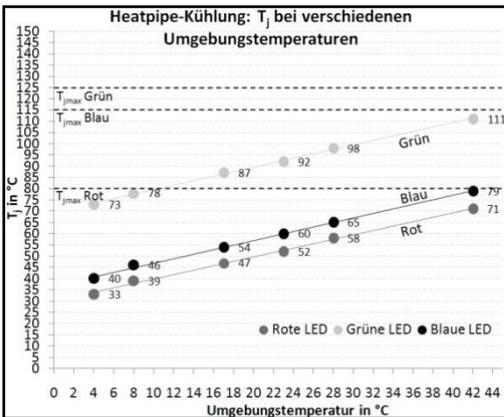


Abb. 9: T_j der LEDs, Heatpipe-Kühlsystem

Des Weiteren wird gezeigt, dass die LEDs des Heatpipe-Kühlsystems bei keiner Umgebungstemperatur die Grenzwerte (T_{jmax}) des Herstellers überschreiten. Dies wird mit der Wasserkühlung nicht ganz erreicht. Während hier die blaue und rote LED bei allen Umgebungstemperaturen ausreichend gekühlt werden, überschreitet die grüne LED den entsprechenden Grenzwert ab einer Umgebungstemperatur von 23 °C, wodurch sich ihre Lebensdauer verkürzt.

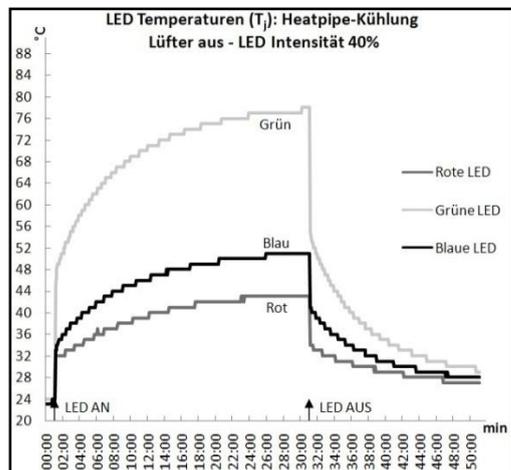


Abb. 11: Temperaturverlauf (T_j), Lüfter aus

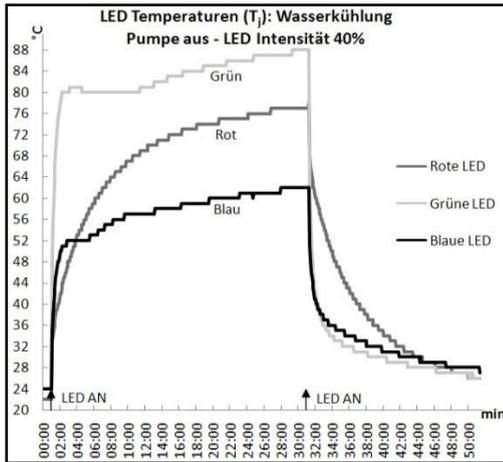


Abb. 12: Temperaturverlauf (T_j), Pumpe aus

Die Abbildungen 11 und 12 zeigen die Sperrschichttemperaturen (T_j) der einzelnen LEDs. Während bei den LEDs des Heatpipe-Kühlsystems Maximaltemperaturen von 43 (rot), 78 (grün) und 51 °C (blau) auftraten, erreichten die LEDs der Wasserkühlung erneut höhere Temperaturen. Diese lagen bei 78 (rot), 88 (grün) und 62 °C (blau).

Die Ergebnisse der Messungen machen deutlich, wie wichtig Lüfter und Pumpe für die jeweiligen Kühlsysteme sind. Die Verlustwärme wird schlechter abgeführt und die Leuchtdioden erwärmen sich stark, wenn die besagten Komponenten ausgeschaltet bleiben. Obwohl bei allen LEDs in diesen Versuchen eine Intensität von 40% eingestellt wurde, erreichten sie bei beiden Kühlsystemen beachtliche Temperaturen. Einen Vergleich der verschiedenen Sperrschichttemperaturen (T_j) bei ein- bzw. ausgeschalteter Komponente des Kühlsystems sowie der entsprechenden LED Intensität zeigt Abbildung 13.

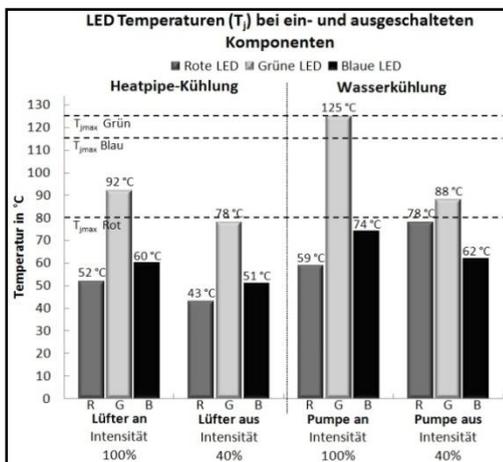


Abb. 13: T_j der LEDs, Komponenten an bzw. aus

Weitere Messungen mit einer höheren Intensität, wie zum Beispiel 60, 80 oder 100 % bei ausgeschalteten Komponenten wurden nicht mehr durchgeführt. Der Grund dafür waren die hohen LED-Temperaturen, die bereits bei einer Intensität von 40% erreicht wurden.

5. ZUSAMMENFASSUNG

In der vorliegenden Arbeit wurden verschiedene Kühlmöglichkeiten vorgestellt, die sich zur Kühlung der Lichtquelle von LED-Projektoren eignen. Dabei wurde die Kühlung mittels eines Heatpipe-Systems und die Wasser- bzw. Flüssigkeitskühlung ins Zentrum der Untersuchung gestellt. In mehreren Messungen unter verschiedenen Bedingungen wurde die Kühlfähigkeit der beiden Kühlsysteme an zwei LED-Projektoren ermittelt, die außer dem Kühlsystem baugleich waren. Beim Vergleich der Messungen konnte gezeigt werden, dass mit dem Heatpipe-Kühlsystem bei allen Versuchen niedrigere LED-Temperaturen als mit der vorliegenden Variante eines Wasserkühlsystems erreicht wurden. Des Weiteren konnte als Ergebnis insbesondere gezeigt werden, dass die Hochleistungs-LEDs des Projektors mit dem Heatpipe-Kühlsystem effizient gekühlt werden, sodass deren Temperaturen deutlich unterhalb der Grenzwerte des LED-Herstellers bleiben. Dies ist eine wichtige Bedingung, um eine hohe LED-Lebensdauer zu erreichen.

6. LITERATURVERZEICHNIS

- [1] Lighting Research Center, Rensselaer Polytechnic Institute. 2003. What is an LED? <http://www.lrc.rpi.edu/programs/NLPIP/lightingAnswers/LED/whatsAnLED.asp>. Letzter Zugriff 17.04.2011
- [2] Deutsche Physikalische Gesellschaft. Zum Glühlampen-Ausstieg. Physik konkret, Nr.8, Deutsche Physikalische Gesellschaft, 2010
- [3] Narendran N., Gu Y. Life of LED-based white. Lighting Research Center, Rensselaer Polytechnic Institute, New York, 2005
- [4] Narendran N., Gu Y. Jayasinghe L., Lalith J., Freyssinier J. and Zhu Y. Long-term performance of white LEDs and systems. Lighting Research Center, Rensselaer Polytechnic Institute, New York, 2007
- [5] Best T., ELEKTRONIKPRAXIS, 2010. Heatpipes kühlen nicht. <http://www.elektronikpraxis.vogel.de/waermemangement/articles/283551/>. Letzter Zugriff: 17.04.11
- [6] eyevis. eyevis new DLP® Rear Projection Series based on LED Light Source. eyevis Reutlingen, 2010.
- [7] Luminus. PhlatLight PT120 Projection Chipset - Summary Data Sheet. Luminus, 2010.

COMPARING DIFFERENT LATENCY-MEASURING METHODS FOR IMAGE TRANSMISSIONS

Denis Merkle
Reutlingen University

denis.merkle@student.reutlingen-university.de

ABSTRACT

This paper describes different methods to measure the latency within image transmissions. All presented methods use high-end sensor hardware and have higher timing resolutions and accuracy than the traditional photo method.

A digital oscilloscope is needed for the numeric measuring method whereas data loggers are sufficient for the shown cross-correlation methods. One cross-correlation method is pulse-based and the other intensity-based. All three methods and their characteristics will be described in this paper.

7. INTRODUCTION

Latencies occur for example, when an user action in a virtual environment needs time until the corresponding frames of the interaction are displayed. These latencies not only depend on the processing speed of the computer, the entire latency is also dependent on latencies in the image transmission, due to different factors: streaming over Ethernet, used cable length, converters and the processing of the display devices. [1][2]

When searching for latency measuring methods for image transmissions in the World Wide Web, there are only few results. But all of them are describing methods to measure the response time of displays by using a side-by-side comparison with a cathode ray tube (CRT) monitor, the photo method. This

measuring method is not very precise, although an author of the website prad.de improved this method by using an optimized stop watch program. Because of its low timing resolution this method does not allow the measurement within the image transmission, for example to measure the latency of pixel-processing equipment with timings < 1 ms. [3]

This paper describes three other methods, which are not based on a side-by-side comparison with a CRT, and finally compares them:

1. Numeric measuring with an oscilloscope
2. Cross-correlation pulsed method
3. Cross-correlation intensity method

The advantage is that these methods measure directly within the image transmission, so they are useable for low latency measurements.

8. DVI-SIGNAL TRANSMISSION

To do low latency measurements within the image transmission it is necessary to understand Digital Visual Interface (DVI) signals, because DVI and the compatible High-Definition Multimedia Interface (HDMI) are the most common used standards today.

Equipped with a high quality oscilloscope it could be possible to measure the Transition Minimized Differential Signaling (TMDS), which is used

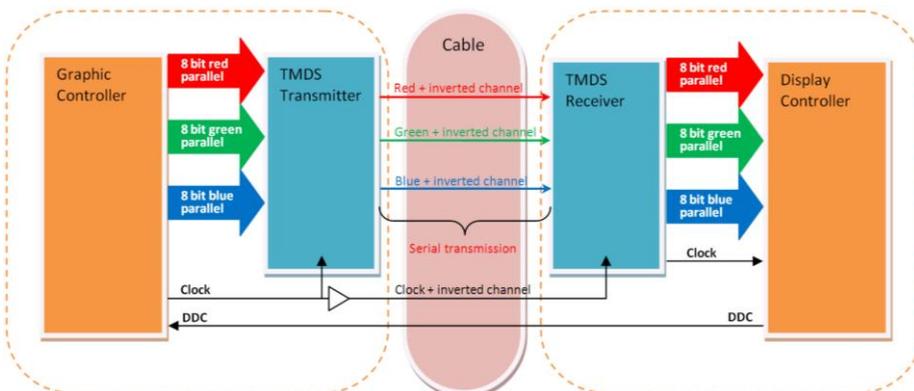


Figure 1 - TMDS transmission

within the DVI cables, directly. In that case the oscilloscope should be able to measure 1.65GHz for the highest possible single-link DVI resolution, because the data is transmitted serially. Figure 1 shows this circumstance. The 24bit color information for red, green and blue are transmitted from the graphic controller to the TMDS transmitter in parallel. For each color there are eight channels. These eight channels get merged in the TMDS transmitter to a serial output which is transmitted on one channel normally and on another inverted. The two transmissions are needed for error corrections due to noise. [4]

The serial bandwidth should be eight times higher than in the parallel DVI signal, but the TMDS protocol adds two additional bits to each pixel for additional error correction and prevention. So the absolute bandwidth is ten times higher than in the DVI signal. [4]

These two additional bits, the *coding bit* and the *DC balancing bit*, make it more difficult to interpret the monitored TMDS signal. There is no possibility to create test patterns on a personal computer, for example via OpenGL, for the video output, which results in defined simple bit patterns in the TMDS signal because the DC balancing bit is switching long patterns with the same state of a bit to guarantee that the electrical ground of the signal, which is the reference for the binary states, will not drift away. The DC balancing Bit is the tenth bit in one series. So if there is a red full screen one pixel on the red channel would be transmitted with "111111100" and the other with "000000001". Additionally, the ninth bit, the coding bit switches the states within one pixel with XOR or XNOR to reduce too many bit changes. As example the 8 bits "01010101" will become "00110011" plus the coding bit which shows, that the bits are coded. If we would only use full colors in our test patterns,

which are normally easy detectable in the signal, such as "11111111", the coding bit and the DC balancing bit would disrupt our constant pattern with a zero (coding bit) and a changing of the whole patterns for from one to zero (DC balancing bit). [4]

With increased complexity and much more expensive hardware it is possible to measure within the TMDS signal. One example can be found in [3].

An easy way to bypass all these problems is to measure the DVI signal after a TMDS receiver. In the DVI signal the data is transmitted in parallel. So the bandwidth per channel will be reduced to the tenth part of the TMDS signal and no additional bits will interfere with the used test patterns. [4]

The rest of the paper will concentrate on how to measure latency in image transmissions within the DVI signal using the previously mentioned methods: numeric measuring, pulsed-based cross-correlation and intensity-based cross-correlation

9. NUMERIC MEASURING

One precise method is the usage of the digital input channels of an oscilloscope. But only high-quality oscilloscopes offer this feature. These oscilloscopes are also equipped with automatic or semi-automatic measurement features, like delay or cursor measurement functions. In contrast better-priced oscilloscopes often do not feature these options, which means that the user has to manually calculate time differences and thresholds of the analog measured digital signal. The next paragraphs refer to the usage of high-quality oscilloscopes with digital input channels to measure latencies in the DVI signal between the output of the graphics board and the input of a display.

The oscilloscope should monitor one bit of a color data channel before and after the devices under test. Figure 2 is showing the test setup. If the same cables for TMDS Receiver1 and TMDS Receiver2

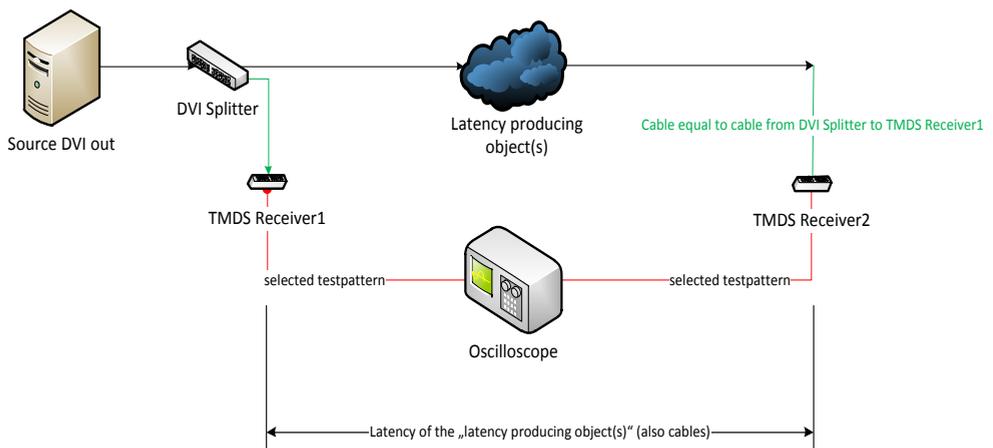


Figure 2 - Test setup

are used, the measured latency will not be affected by the test equipment.

There are many ways to design full-screen test patterns for this purpose. These are video patterns which use the whole resolution of the display, so that no undesired pixel appears on the screen. One of the simplest patterns could be a black background without foreground information in the first frame and in the second frame one colored pixel which could be used as a trigger. The measured latency will be the onset delay of this pixel between TMD5 Receiver1 and TMD5 Receiver2. If the latency of your latency-producing objects is content-based, for example when the image is streamed over Ethernet by a codec with temporal compression between the frames, then you have to create fitted test patterns to simulate the scenario that should be measured.

When the image changes from the first to the second frame of the test pattern, the oscilloscope will perform a single trigger event and present the results on the screen.

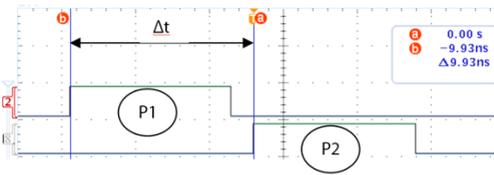


Figure 3 - Oscilloscope screenshot

Figure 3 illustrates such an oscilloscope screenshot. In this example, the measuring was done by the cursor measuring feature of the oscilloscope. “P1” is the active test pixel of TMD5 Receiver1 and “P2” is the corresponding pixel on TMD5 Receiver2. “Δt” is showing the resulting latency as difference between the two onsets of the pixel on the two different TMD5 receivers. In this example of the Tektronix MSO 4054 oscilloscope, the latency is shown in the upper right box as “Δ”.

Table 4 - Deviation

Time domain	Deviation between max and min result of 30 measurements (timing resolution 60,6 ps)
0ns – 6ns	35%
6ns – 20ns	5%
20ns – 80ns	3%
80ns – 120ns	2%
>120ns	1%

Table 4 shows the deviation between the maximum and the minimum result of several test series with 30 single measurements in different time domains. This data is based on measurements with a Tektronix MSO 4054 and the rest of the test setup consists of some Molex DVI cables and TMD5 receivers from Silicon Image (SiI161BCT100). The deviation

between 0 ns and 6 ns is with 35% a bit high, but in arithmetic mean the results are as expected for the used cable length [5]. These results were achieved with a series of tests with copper and optical fibre DVI cables to verify and evaluate the stability of the numeric measuring.

This means that the results have to be averaged over a series when measuring in time domains down to 80ns or lower with this test equipment, because the deviation is too high to rely on a single measurement.

10. CROSS-CORRELATION

Numeric measurement functions well whenever the onset is clearly detectable. But when the onset could not be measured digitally, a threshold must be set to detect the onset. In some cases this is acceptable, for example when the increases of the onsets on both signals are equal. But when there are different analog or noisy signals it is impossible to set a correct threshold. Figure 4 is showing a perfect digital, a noisy and an analog signal. Only in the perfect signal the onset is exactly detectable.

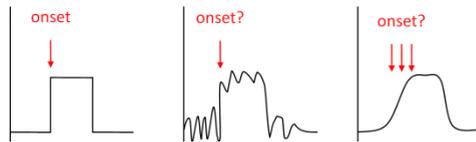


Figure 4 - Onsets in different signals

An easier way to do the measurement is to use a method which doesn’t need additional determined conditions. The measurement via cross-correlation achieves this requirement. For the use of cross-correlation methods the only requirement is to save the signals from the initial and final point as discrete values. This could be managed by high class oscilloscopes, but also with better-priced data loggers. The signals for the cross-correlation methods are monitored similarly to the numeric measuring method on the parallel DVI signals and not within the serial TMD5 signals.

In simple words, the cross-correlation shifts one diagram over the other and compares them. For latency measurements, it is not relevant the degree of correlation but the shift difference between both diagrams when the highest correlation is reached.

$$corr\{k\} = \frac{\sum_{i=1}^N x[i] * y[i + k]}{\sqrt{\sum_{i=1}^N (x[i])^2 * \sum_{i=1}^N (y[i + k])^2}} \quad (1)$$

The cross-correlation formula for discrete values (1) is calculated over the sum. “x” and “y” are the both signals, “i” is the actual time position in the signal and “k” is the shift value. The denominator only normalizes the correlation results to values between -1 and 1. [6][7][8]

Potential key results:

- 1: completely correlated with inverse signals
- 0: uncorrelated
- 1: completely correlated

All other results between those values are describing the degree of correlation.

One positive aspect of the cross-correlation method is the fact that it does not require a high degree of correlation. The two input signals do not need to be equal. If one or even both signals include a lot of noise, the degree of correlation will not be as high as 100% but the resulting latency will be the same. [8]

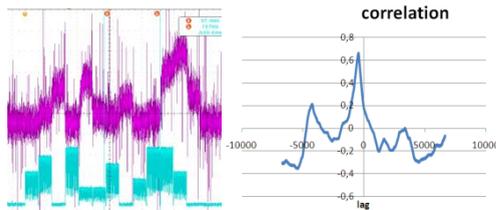


Figure 5 - Noisy signal

Figure 5 shows an example of a noisy signal. The membership between the intensity steps in the signals is visible for humans and the cross-correlation function recognizes these steps, too. In this example the degree of correlation is as high as 67% and this peak is not roughly reached in other time shifted constellations.

Cross-correlation offers new potentials compared to numeric measuring. For example sensitive photo sensors with a high degree of noise could be easily used without being worried about setting thresholds.

10.1 PULSED METHOD

The test setup of a cross-correlation pulsed method measurement in DVI signals looks like the setup for numeric measuring shown in Figure 2. Only the oscilloscope could be replaced by a data logger. The measuring points are also located at the same digital data channels of one color bit. Under most test conditions it is useful to measure with analog probes of the data logger or oscilloscope, whose range of application is much higher than the one of digital probes.

The test patterns for the here processed example of measurement within the DVI signal also looks like the test patterns of the numeric measuring. The difference is a sequence of red colored pixels compared to only one red pixel. This sequence should be created without repetitions in it, so that all parts of the sequence will be unique.

Figure 6 shows such a unique test pattern signal. If the sequence is not created with this unique precon-

dition the cross-correlation could produce several high results which would eventually result in a wrong automatic calculated latency. It is also possible to use only one red pixel in the test pattern, but this can cause higher deviations which must be compensated with more measurement passes.

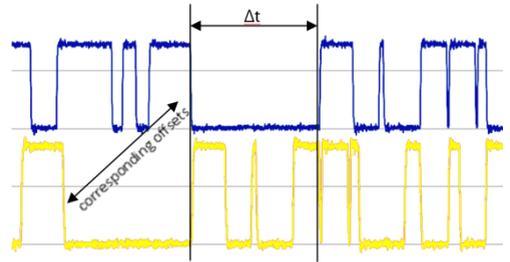


Figure 6 - Clipping of pulsed cross-correlation test pattern signal

10.2 INTENSITY METHOD

The cross-correlation intensity method complements the pulsed method. So they are not opposed to each other.

The idea is to improve the pulsed method by adding additional states. This is helpful if the cross-correlation method is used with analog signals like one from a photo sensor which is added to the brightness or other signals which are not >90% identical to each other. To realize several different states in digital channels they have to be combined by a digital analog converter (DAC).

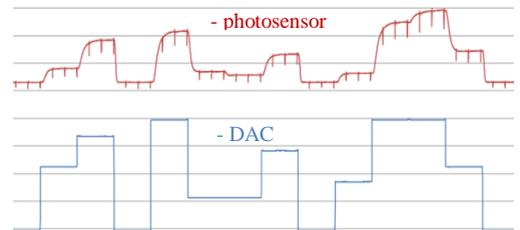


Figure 7 - Intensity on photo sensor and DAC

The test patterns should contain values of channels which are used in the digital analog converter. For example these could be the three highest bits of a color. The signals on the measuring points will correspond proportionally to each other. This is presented in Figure 7 by an example of the intensity method with a photo sensor (red graph) and the DAC output on the blue graph. Each state in this figure represents one frame. The small spikes on the photo sensor graph are showing the pulsed backlight with a frequency of ~200Hz. Some optimization effects, like *Overdrive*, from the display manufacturer can also be appreciated when two consecutive frames have the same state.

Figure 8 shows a test setup for display latency measurements with the cross-correlation intensity

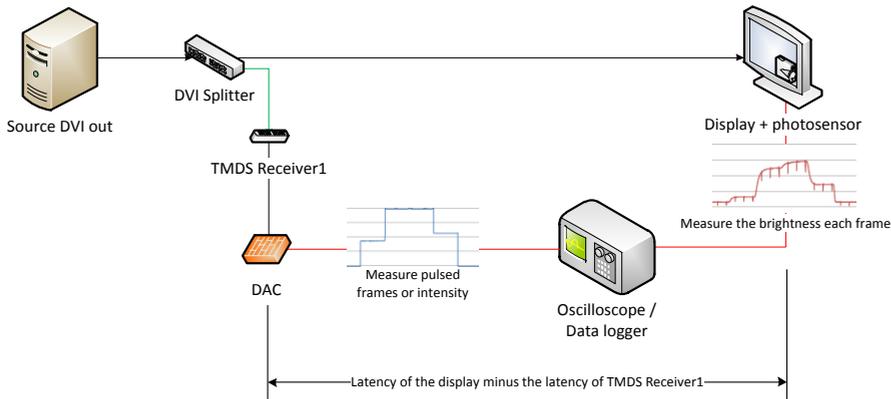


Figure 8 - Display latency setup

method. In this setup the latency of the display will be measured without the latency of the TMDS Receiver1 but there is an additional latency of the photo sensor. TMDS receivers have latencies of about 100 ns. So it could be neglected with display respond times >1 ms. The additional added latency of the photo sensor is 10 μ s, that is negligible, too.

If the intensity method is used to measure fast signals, some few pixels for example, a fast digital to analog converter should be selected. If the DAC is to slow, because of its capacitive behavior, the test patterns should be changed so that lines instead of pixels could be measured. Then the DAC has enough time to get on the exact state.

10.3 DATA ANALYSIS

In contrast to the real-time results of the numeric measuring, the cross-correlation methods need either a very expensive logic analyzer or a computer which calculates the cross-correlation function for the two input signals.

Implemented in Visual Basic for Applications and Microsoft Excel it is possible to combine the calculation of the latency with detailed statistics and graphs for documentation or presentation purposes.

The timing resolutions of the cross-correlations methods are also dependent on the used test equipment. With the used Tektronix MSO4054 a timing resolution of 0.4 ns can be reached.

11. ANALYSIS

To evaluate these methods the latencies of several copper and optical fiber cables were tested.

The following cables were used for the evaluation:

- copper:
 - Molex Duallink 2m, 5m, 10m
 - Eyevis 30m EXT (Extender)
- Optical fiber with Eyevis EY-201-TR (multimode with SC-SC connectors)

As it can be seen on Figure 9, nearly two-thirds of the speed of light in vacuum can be reached with optical fiber cables. The decrement in speed with cables smaller than 10 m is affected by the latency of the optical converters. The speed of the copper cables is a little bit higher. Only the extender within the 30 m cable added some additional latency.

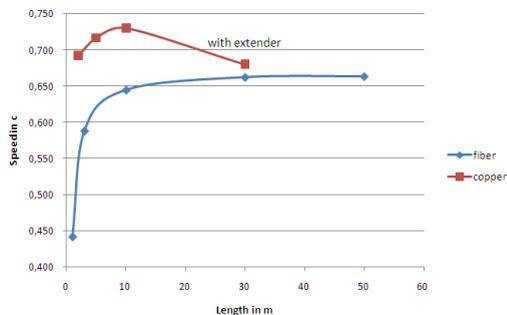


Figure 9 - DVI cable speed

All these results are within common measured and calculated speeds for copper and fiber optic cables under consideration of indices of refraction and the permittivity of the cable dielectric [9][5][10]. The results of Figure 9 were measured with the numeric measuring method.

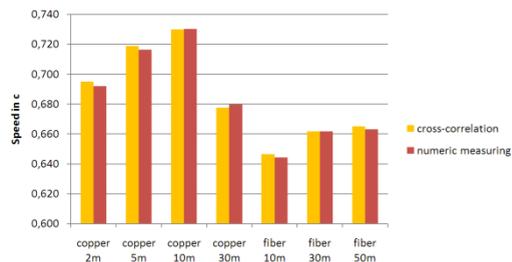


Figure 10a - Comparing latencies measured by pulsed cross-correlation and numeric measuring

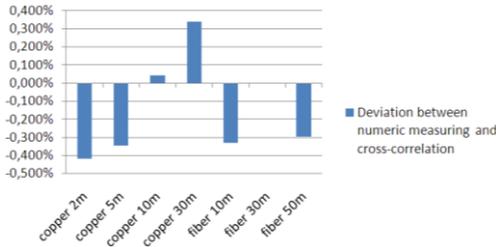


Figure 10b - Comparing latencies measured by pulsed cross-correlation and numeric measuring

In contrast the results of the pulsed cross-correlation method are shown in Figure 10a/b. The deviation of the results is smaller than 0.5% in both, the positive and the negative directions. So this difference is insignificant and it can be asserted that both methods have the same accuracy on the same test equipment.

12. CONCLUSION

In summary both basic methods, the numeric and the measuring with cross-correlation, work accurately when high class test equipment is used.

Table 5 - Characteristics

	Numeric measuring	Cross-correlation pulsed	Cross-correlation intensity
Speed	fast	Need offline analysis	Need offline analysis
Discrete values	✓	✓	✓
Noisy signals	Not accurate	✓	✓
Analog	Need thresholds	Dependent on test pattern and setup	✓
Cost	Need high class oscilloscope	Only better-priced data logger needed	Only better-priced data logger needed

Table 5 shows the characteristics of the different methods. Speed is an advantage of the numeric measuring, whereas the cross-correlation methods need more time for copy and calculation of the data. Digital signals could be measured with all three methods without any complications. When there is noise on the signals, mostly in analog signals, the cross-correlation methods have no problems to calculate the latency. When analog signals should be measured it is easy to use the cross-correlation methods, because no threshold has to be set for on- or offsets. Finally the cross-correlation methods could be set up cheaper than numeric measuring.

If a fast or first measuring without successive documentation with charts should be done, then numeric measuring has an advantage. But when charts for documentations were needed the discrete values

have to be saved and the speed advantage of numeric measuring is getting lost.

If new test equipment is needed for the latency measurements both cross-correlation methods could be realized with better-priced data loggers. These methods are well-priced in contrast to the expensive high class oscilloscope which is needed for numeric measuring. But there are two further advantages for the cross-correlation methods. It is easy to measure noisy and analog signals without being worried about setting thresholds, using filters, amplifiers, etc.

In summary the advantages of the cross-correlation methods outweigh the advantages of numeric measuring when using a basic spreadsheet macro.

13. REFERENCES

- [1] Brunnett, G. and Coquillart, S. and Welch G. 2008. Virtual Realities: Dagstuhl Seminar 2008. Springer.
- [2] Schürmann, B. 2004. Grundlagen der Rechnerkommunikation: technische Realisierung von Bussystemen und Rechnernetzen. Vieweg + Teubner.
- [3] Thiemann, T. PRAD. <http://www.prad.de/new/monitore/specials/inputlag/inputlag.html> [25.03.2011]
- [4] DDWG Digital Display Working Group. 1999. Digital Visual Interface DVI. Specification Version 1.0. http://www.ddwg.org/lib/dvi_10.pdf [25.03.2011]
- [5] Keller, A. 2011. Breitbandkabel Und Zugangsnetze: Technische Grundlagen Und Standards. Springer.
- [6] Burger, W. and Burge, M.J. 2006. Digitale Bildverarbeitung. Springer.
- [7] Kreuder, F. 2008. 2D-3D-Registrierung mit Parameterkopplung für die Patientenlagerung in der Strahlentherapie. KIT Scientific Publishing.
- [8] Udayashankara, V. 2010. Real Time Digital Signal Processing: Fundamentals, Algorithms and Implementation Using TMS Processor. PHI Learning Pvt. Ltd.
- [9] Brückner, V. 2003. Optische Nachrichtentechnik: Grundlagen und Anwendungen. Vieweg + Teubner.
- [10] Weißgerber, W. 2007. Elektrotechnik für Ingenieure. Vieweg + Teubner

Virtuelle Realität bei Medien- und Kommunikationsinformatikern

Patrick Gaßner, Samet Keser, Julian Doll,
Christoph Meißner, Emil Wiese, Erik Herrmann
VRlab – Master- Medien- und Kommunikationsinformatik Hochschule Reutlingen
vrlab@hochschule-reutlingen.de

ABSTRACT

In diesem Beitrag wird das Masterprojekt VRlab des Studiengangs Medien- und Kommunikationsinformatik der Hochschule Reutlingen vorgestellt. Das „Virtual Reality Laboratory“ beschäftigt sich hauptsächlich mit Forschungsfragen rund um die Mensch-Maschine-Interaktion in virtueller Umgebung.

1. MASTERPROJEKT VRLAB

Fester Bestandteil des Master-Studiums der Medien- und Kommunikationsinformatik ist das über zwei Semester dauernde, iterativ aufgebaute Masterprojekt. Es bietet den Studierenden einen inhaltlichen Rahmen und lässt ihnen zugleich große Freiheit bei der Wahl ihrer Vertiefungsbereiche in den Themengebieten des Projektes. Durch die aktive Arbeit in einem Projektteam sowie die Zusammenarbeit mit Studierenden anderer Semester werden ständig neue Herausforderungen an die soziale Kompetenz der Teammitglieder gestellt. Der Umgang mit Konflikten, die Präsentation von Projektergebnissen und die Kommunikation innerhalb des Teams als auch mit den betreuenden Dozenten tragen dabei zur Entwicklung der „soft skills“ der Studierenden bei. Das VRlab ist dabei eines von drei Projekten, in denen sich die Studierenden über einen längeren Zeitraum intensiv und vertiefend mit einem Themenbereich der Medien- und Kommunikationsinformatik auseinandersetzen. Entwicklung virtueller Umgebungen und die Interaktion zwischen Mensch und computergenerierter Realität bilden dabei die Eckpunkte des VRlab [1].

2. DAS VRLAB

Seit dem Wintersemester 2007/08 betreiben die Teammitglieder ein eigenes Labor mit moderner Hardwareausstattung für Forschung und Entwicklung. Herzstück und zugleich räumlicher Mittelpunkt des Labors bilden die sogenannte Blackbox und ihr Interaktionsraum (Abbildung 1). Die Blackbox dient als Rückprojektionssystem für stereoskopische Bildausgabe unter Einsatz der Polarisationsfiltertechnik in einer Größe von 125 Zoll. Im Interaktionsraum vor der Blackbox können räumliche Systemeingaben wahlweise durch ein kommerzielles oder ein eigenentwickeltes Tra-

ckingsystem erfolgen. Zudem sind weitere Stereo-Projektionssysteme mit Interferenzfilter- und Shutter-Technik vorhanden. Über das Engagement der Projektmitglieder hinaus, stellt das VRlab ständig Themen zur Bearbeitung als Bachelor- oder Master-Thesis zur Verfügung.



Abbildung 1: Head-Tracking-Anwendung

3. DIE PROJEKTE

Im Gegensatz zu den anderen MKI-Masterprojekten ist das VRlab kein in sich geschlossenes Projekt, sondern versteht sich vielmehr als räumlicher und inhaltlicher Rahmen, der mehrere Teilprojekte aus dem Themengebiet der virtuellen Realität in sich vereint. Nachfolgend werden aktuell laufende Projekte vorgestellt.

3.1 LOW-COST-TRACKING

Kommerzielle optische Trackingsysteme sind teuer in der Anschaffung und aufwendig in Montage und Inbetriebnahme. Aus diesem Grund wird im VRlab an der Entwicklung eines aktiven Low-Cost-Tracking-Systems gearbeitet. Zwei stereoskopisch ausgerichtete herkömmliche USB-Webcams mit Infrarot-Bandpassfilter kommen hierbei zum Einsatz. Mit Hilfe der Kalibrierungsmethoden aus der OpenCV-Bibliothek erfährt das Tracking-System eine - im Idealfall - einmalige Kalibrierung. Mit aktiven Infrarotmarkern lassen sich mehrere Markerbewegungen im Raum ohne Verdeckung verfolgen. Die Routinen für die Kalibrierung, das Blob-Erkennung und die 3D-Koordinatenberechnung sind vom VRlab-Team konzipiert, implementiert

und funktionsfähig. Ausstehend ist lediglich die Entwicklung der VRPN¹-Schnittstelle zur Anbindung an eine Anwendung. [2][3]

3.2 VRLAB-FRAMEWORK

Das Konzept des Frameworks wurde aufgestellt um einerseits die in fast allen Projekten des VRLabs befindlichen Gemeinsamkeiten auszulagern und andererseits um Modularisierung und somit Transparenz zu erreichen. Die meisten der im VRLab aktiven Projekte realisieren eine Benutzer-Eingabe, ähnlich einer Maus-Eingabe. Der Multi-Touch-Tisch, der Eyevis-Cube (siehe Kapitel 3.3) sowie das iPad ermöglichen eine Eingabe mehrerer(x,y)-Koordinaten zur selben Zeit, die Tracking-Systeme auch die Erfassung von räumlichen Positionsangaben.

Realisierbar wäre das Herauslösen und Mitnehmen der Eingabemöglichkeit aus dem aktuell bedienten Gerät, also die räumliche Trennung von Eingabe, Verarbeitungspunkt und der Ausgabe. Die Eingabe am Multi-Touch-Tisch könnte dann beispielsweise im Zuge einer Präsentation oder Anwendung auf ein iPad übertragen werden und via Funknetzwerk - ohne den Anwendungsfluss zu unterbrechen - mitgenommen werden.

Die sich ergebenden Synergie-Effekte sind hierbei:

- Transparenz der Eingabe gegenüber der Anwendung
- damit verbunden die Transparenz und Modularität der Software (-schichten) im Sinne der OOP-Entwicklung und Architektur und der Wiederverwendbarkeit des Quellcodes.
- Entlastung der Entwicklung neuer Anwendungen in dem auf bereits existierenden Quellcode zurückgegriffen werden kann
- durch Netzwerkübertragung der Eingabe wird gleichzeitig eine Abstraktion der verwendeten Programmiersprachen - und generell - der Technologie möglich
- der Freiheitsgrad des Anwenders bei der Anwendung wird verbessert
- die Studierenden des VRLab-Teams lernen den Umgang mit einer integrativen Lösung und üben ihre Projekte an gemeinsamen, verbindlichen Richtlinien auszurichten

3.3 MULTI-TOUCH

Seit Beginn des Wintersemesters 2010 arbeitet das VRLab-Team auch an zwei Multi-Touch-Systemen. Eines davon ist ein Rückprojektionssystem mit

vertikaler Bildschirmfläche. Im VRLab trägt dieses System aufgrund seiner würfelförmigen Form den Namen Cube, siehe Abbildung 2. Die Eyevis GmbH aus Reutlingen, einer der führenden Hersteller von Großbildsystemen, stellte den Cube der Hochschule zu Forschungszwecken kostenfrei zur Verfügung. Der Rückprojektioncube diente bei Übergabe an die Hochschule lediglich als Ausgabegerät. Die Aufgabe des VRLab-Teams besteht darin, den Cube unter Verwendung von Low-Cost-Komponenten zu einem Multi-Touch-Gerät zu erweitern. Die für den Cube benötigten Bauteile werden ebenfalls von Eyevis bereitgestellt.

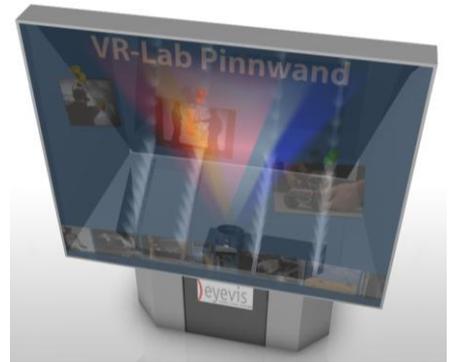


Abbildung 2: Multi-Touch-Cube

Die Umsetzung der Multi-Touch-Funktionalität wird unter Anwendung des Rear-Diffuse-Illumination-Ansatzes realisiert [4]. Hinter der Projektionsfläche angebrachte Infrarot-LEDs leuchten hierbei den Innenraum des Cubes möglichst homogen aus. Eine Kamera mit hoher Auflösung und hoher Bildwiederholrate erfasst mittels eines eingebauten Infrarot-Bandpassfilters nur das von den Fingern diffus reflektierte Infrarotlicht und erkennt daraus eine Berührung der Projektionsfläche [5][6].

Dasselbe Verfahren wird auch bei dem zweiten System, dem Multi-Touch-Tisch angewendet. Der einzige Unterschied besteht darin, dass es sich hierbei um eine horizontale Touch-Fläche handelt und die Hardwarekomponenten von der Hochschule bereitgestellt werden.

Die Entwicklung von Anwendungen für beide Systeme steht bisher noch aus, soll jedoch in den kommenden Semestern verstärkt im Vordergrund stehen.

3.4 VRLAB-KINECT

Neben der Entwicklung neuer Interaktionsmöglichkeiten beschäftigt sich das VRLab mit der Untersuchung und Evaluierung neuer Technologien aus der Unterhaltungsindustrie auf ihre Anwendungsgebiete im Informatikumfeld. Aktuell wird dabei an einer

¹ VRPN = Virtual Reality Peripheral Network

Schnittstelle zwischen der interaktiven und markerlosen Bewegungssteuerung Microsoft Kinect und einer medizinischen Software gearbeitet, mit dem Ziel, Erkenntnisse über mögliche Einsatzgebiete der Gestensteuerung im medizinischen Umfeld zu gewinnen. Hierbei werden die Körperbewegungen des Nutzers anhand eines Infrarot-Tiefensensors und einer RGB-Farbkamera erfasst und mittels bekannter Bildverarbeitungsverfahren selektiert. Durch definierte Merkmale der selektierten Bewegungen ist es möglich, zwischen verschiedenen Gesten zu unterscheiden. Für die Interaktion mit der Medizinsoftware werden zunächst nur Gesten für Zoom und Rotation eingesetzt.

3.5 ANDROID VRLAB-REMOTE CONTROL

Das jüngste Beispiel für die Vergabe und Bearbeitung einer Bachelor-Thesis im VRLab ist das Projekt „Android VRLab-Remote Control“. Die bereits auf einem PDA-System implementierte Fernsteuerung des Licht- und Audiosystems sowie der Maus- und Beamersteuerung des VRLab wurde in der Bachelor-Thesis von Michael Rudolf auf das mobile Betriebssystem Android portiert und ihr Funktionsumfang erweitert. Die drahtlose Kommunikation zwischen mobilem Gerät als Client und dem lokalen Rechner als Server erfolgt über W-LAN. Über die serielle Schnittstelle des Servers werden die beiden Beamer angesteuert, die Regelung der Lichtanlage erfolgt über das DMX²-Protokoll. Audio- und Maussteuerung werden direkt auf das Serversystem übertragen [7].

4. FAZIT

Das Masterprojekt VRLab ist aufgrund seiner vielfältigen Teilprojekte eine beliebte Wahl bei den Master-Studenten. Dies belegt auch die aktuelle Anzahl an Projektteilnehmern. Mit 11 Studierenden im Sommersemester 2011 erreicht das VRLab die bislang höchste Teamgröße, was nicht zuletzt an den breit gefächerten Aufgabenbereichen liegt.

Nicht nur Forschung und Entwicklung, sondern auch handwerkliche Tätigkeiten bei der Erweiterung und Instandhaltung der Infrastruktur sowie redaktionelle und organisatorische Aufgaben bei öffentlichen Veranstaltungen sind Teil der Aufgaben im VRLab. Dass das Masterprojekt VRLab nicht nur für die Reutlinger Master-Studenten sehr interessant sein kann, zeigt das Engagement zweier Austauschstudenten, die in diesem Semester ein internationales VRLab-Projektteam formen. Eine solche Zusammenarbeit mit ausländischen Studierenden findet in diesem Semester zum ersten Mal statt, aber auch in Zukunft soll auf eine internationale Beteiligung am Masterprojekt Wert gelegt werden.

5. LITERATURVERZEICHNIS

- [1] Hochschule Reutlingen. 2011. Modulhandbuch Teil B, Medien- und Kommunikationsinformatik (MKI) – Master. <http://www.inf.reutlingen-universi.de/fileadmin/downloads/modulhandbuch/Modulhandbuch-Teil-B-MKI-Master.pdf>. Letzter Zugriff 18.04.2011
- [2] OpenCV. 2011. Open Source Computer Vision. <http://opencv.willowgarage.com/wiki>. Letzter Zugriff 19.04.2011
- [3] VRPN. 2011. VRPN 07.28. <http://www.cs.unc.edu/Research/vrpn/index.html>. Letzter Zugriff 18.04.2011
- [4] Andrew D. Wilson: TouchLight: an imaging touch screen and display for gesturebased interaction, in Proc. of the 6th international conference on Multimodal interfaces, PA, Seiten 69-76, ACM, New York, 2004
- [5] NUI Group Community. 2011. NUI Group – Natural User Interface Group. <http://nuigroup.com/>. Letzter Zugriff 19.04.2011
- [6] NUI Group Community. 2011. Community Core Vision, <http://ccv.nuigroup.com/>. Letzter Zugriff 19.04.2011
- [7] Rudolf, M. 2011. Umsetzung einer Smartphone-Mediensteuerung für das VRLab. Hochschule Reutlingen. Medien- und Kommunikationsinformatik.

² DMX = Digital Multiplex

Web * Mobile * Content * Collaboration

Benjamin Blubacher

Hochschule Reutlingen

benjamin.blubacher@

student.reutlingen-university.de

Nico Hornung

Hochschule Reutlingen

nico.hornung@

student.reutlingen-university.de

ABSTRACT

Der Begriff „Computer Supported Collaborative Work“ beschreibt ein Forschungsgebiet der Informatik bei dem es sich um computerbasierte Gruppenarbeit handelt. Die vorliegende Arbeit repräsentiert das Masterprojekt Web Mobile Content Collaboration oder kurz WMC². Dabei wird anhand einer Beispielarchitektur ein Portalsystem entwickelt. Mit verschiedenen Geräten wie Notebooks, Tablet-Computer oder Mobiltelefonen soll mit dem System gearbeitet werden. Das System soll es den Studenten und Mitarbeitern der Hochschule Reutlingen ermöglichen, innovativer und effektiver zusammen zu arbeiten.

1. EINLEITUNG

Der Zugriff verschiedener Personen auf gemeinsam genutzte heterogene Informationen stellt ohne die Hilfe von unterstützender Software ein Problem dar. Der Begriff „Kollaboratives Arbeiten“ fasst die wichtigen Bereiche wie die Kommunikation unter den einzelnen Parteien, das Bearbeiten von gemeinsamen Artefakten sowie die Koordination des Teams zusammen. Um die Gruppenarbeit zu vereinfachen werden kollaborative Systeme - oder auch CSCW³⁴-Systeme - benötigt, die ein kollaboratives Arbeiten ermöglichen [1].

2. DAS PROJEKT

Ein wichtiger Bestandteil des Masterstudiengangs Medien- und Kommunikationsinformatik (MKI) der Hochschule Reutlingen ist das Masterprojekt. Ziel dieses Projekts ist es, sich über einen Zeitraum von zwei Semestern mit wissenschaftlichen Themen auseinanderzusetzen und diese gemeinsam zu bearbeiten. Eines dieser Projekte befasst sich mit der Thematik einer zentralen Plattform für kollaboratives Arbeiten, teilen von Informationen sowie dem Zentralisieren von Arbeitsabläufen. Hierzu soll ein Portalsystem entworfen werden. Dieses wird dabei auf die vorhandenen Dienste der an der Hochschule Reutlingen verwendeten Systeme zurückgreifen. Des Weiteren ist geplant das System um zusätzliche Funktionalitäten zu erweitern. Dabei

sollen Dienste gängiger CMS³⁵, DMS³⁶ und CSCW-Systeme integriert werden. Um das Portal auch für zukünftige Anforderungen kompatibel zu machen wird ein modularer Aufbau beabsichtigt. Zur Präsentation der einzelnen Arbeitsbereiche werden individuelle Darstellungsformen für Mobil- und Webanwendungen entworfen.

2.1 UMFRAGE

Um Aufgaben zu priorisieren und neue Anforderungen einzuholen wurde eine Online-Umfrage an der Hochschule Reutlingen durchgeführt (im Zeitraum 03.01.2011 bis 16.01.2011).³⁷ Insgesamt nahmen 587 Studenten (13,7% der Gesamtzahl an Studenten) und 107 Mitarbeiter (16,3% der Gesamtzahl an Mitarbeitern) der Hochschule Reutlingen teil.³⁸

Als wichtigen Bestandteil einer kollaborativen Software sehen die meisten eine asynchrone Kommunikation über E-Mail, Foren und Blogs. Zudem wird das Bereitstellen und das gemeinsame Bearbeiten von Dateien und Dokumenten als essentiell angesehen.

Von einem großen Teil werden soziale Netzwerke wie Facebook oder XING (79% der Studenten) und Internet-Telefonie wie Skype (77% der Studenten) privat benutzt.

Über die Hälfte aller Teilnehmer sieht die Nutzung des Portals über ein Smartphone als "wichtig" oder "sehr wichtig" an. Dem Großteil der Mitarbeiter ist es "wichtig" (26%) bzw. "sehr wichtig" (36%) ortsungebunden im Team arbeiten zu können. Die aus der Online-Umfrage resultierenden Anfor-

³⁵ CMS – Content Management System

³⁶ DMS – Dokumenten Management System

³⁷ Die genauen Ergebnisse der Umfrage sind aufgrund persönlicher Kommentare nicht öffentlich. Bei Fragen nehmen Sie bitte Kontakt mit uns auf. [2]

³⁸ Die Hochschule Reutlingen hat derzeit (Stand: März 2011) ca. 4300 Studenten und ca. 665 Mitarbeiter (setzt sich zusammen aus ca. 135 Professoren, ca. 250 Lehrbeauftragten und ca. 280 Hochschulmitarbeitern). [3]

³⁴ CSCW - Computer Supported Collaborative Work

derungen sind in der nachfolgenden Tabelle aufgelistet.

Tabelle 1: Wichtigste Anforderungen der Studenten und Mitarbeiter an das Projekt

Studenten	Mitarbeiter
<ul style="list-style-type: none"> • Dateien gemeinsam bearbeiten • Einbeziehung von mobilen Endgeräten • Filter (z.B. Spamfilter für Mails) • Synchronisation • SMS-Benachrichtigungen • Anpassbarkeit, Individualisierbarkeit 	<ul style="list-style-type: none"> • Versionierung • Einbeziehung von mobilen Endgeräten • einfache Handhabung • selbsterklärend • Online-Präsentationen • persönlicher Kalender

2.2 ÄHNLICHE PROJEKTE

„Tiki Wiki CMS Groupware“ [4] ist ein freies CMS um Portale ähnlich den Vorstellungen von WMC² zu erstellen. Das „Liferay Portal“ [5] ist eine Open-Source-Software die als Enterprise-Portal eingesetzt wird und nebenläufige Benutzer auf einem Server unterstützt.

Die Aufgabenstellung von WMC² wurde auch von Masterstudenten der Pädagogik, Psychologie und Soziologie an der TU Darmstadt aufgegriffen. Entstanden ist das Projekt „MyPaed“. [6]

3. UMSETZUNG

Die Teams innerhalb des Projekts teilen sich auf die drei nachfolgenden Gebiete auf.

3.1 ARCHITEKTUR

Als Architekturvorlage des Projekts dient eine Referenzarchitektur des Fraunhofer Instituts für Arbeitswirtschaft und Organisation (IAO).

Diese ist Bestandteil der PADEM (Portal Analyse und Design Methode), einem Vorgehensmodell für Portalprojekte. Diese Methode beruht auf einem klassischen Phasenmodell des Software Engineerings, das die fünf Phasen Strategie, Analyse, Konzeption, Realisierung und Einführung umfasst [7].

Die Referenzarchitektur ist Teil der Konzeptionsphase, liefert die Basis für die Kriterien der Marktübersicht und ermöglicht eine produktneutrale Soll-Modellierung [7].

Diese wird mit Webservices realisiert, deren Schnittstellen als XML definiert sind und so unabhängig von der Programmiersprache kommunizieren können. Auf Präsentationsebene wird dabei der Schwerpunkt auf Webbrowser sowie mobile Endgeräte gelegt. Für das Single-Sign-On, also das einmalige Anmelden an der Webanwendung zur Nutzung von weiteren Anwendungen, soll das XML-Framework SAML (Security Assertion Mar-

kup Language) zum Austausch von Authentifizierungs- und Autorisierungsinformationen getestet werden. [8]

Daraus ergeben sich folgende Forschungsfragen die im Laufe des Projekts WMC² beantwortet werden sollen:

- Ist die Referenzarchitektur des Fraunhofer Instituts für das Projekt geeignet?

- Lässt sich diese Architektur überhaupt praktisch umsetzen?

3.2 WEB CLIENT

Auf Clientseite wird eine Webanwendung realisiert, wodurch die Interaktion mit dem Benutzer über einen Webbrowser stattfindet. Dies hat gegenüber herkömmlichen Client-Server-Anwendungen den Vorteil der Plattformunabhängigkeit sowie den geringeren Hardware-Anforderungen an das Endgerät selbst.

Als Framework für den Webclient dient das Google Web Toolkit (GWT). [9] Dabei handelt es sich um einen Satz von Entwicklungswerkzeugen mit denen Rich Internet Applications erstellt werden können. Der serverseitige Quellcode wird dabei in Java geschrieben und später vom Compiler in JavaScript umgesetzt [10]. Für mobile Endgeräte bietet das GWT Mobile Webkit [11] Schnittstellen um HTML5 und WebKit Funktionen zu unterstützen und in Browsern der Betriebssysteme iOS und Android darzustellen [11].

3.3 MOBILE CLIENT

Es stellt sich die Frage, warum die oben beschriebene Website nicht auch auf mobilen Plattformen wie z.B. einem Mobiltelefon oder einem Tablet-Computer eingesetzt wird, da die meisten dieser Geräte ebenfalls über einen Webbrowser verfügen. Hierbei sollten Vor- und Nachteile für die Umsetzung einer nativen Anwendung sowie den Einsatz einer Website betrachtet werden.

Beide Lösungen haben Vorteile die für beide Anwendungsarten gelten, wie z.B. die Anzeige von Informationen und interaktiven Inhalten. Abgesehen davon gilt bei der Umsetzung: Was bei der einen Anwendung die Vorteile, sind bei der Anderen die Nachteile und umgekehrt. So hat die native Anwendung vollen Zugriff auf alle Hardwaremodule sobald eine Schnittstelle³⁹ vorhanden ist, was bei einer webbasierten Anwendung nur bedingt oder gar nicht der Fall ist. Da die Anwendung lokal auf dem Gerät ausgeführt wird kann diese auch ohne eine bestehende Internetverbindung ausgeführt werden. Ist keine Verbindung via Wireless-LAN

³⁹ engl. Application Programming Interface (API)

oder Mobilfunknetz (UMTS⁴⁰ usw.) vorhanden, kann die webbasierte Anwendung nicht verwendet werden. Außerdem sollte man in Betracht ziehen, dass Daten verloren gehen können sobald die Verbindung zum Internet abbricht.

Das Problem bei der Verwendung von mobilen nativen Anwendungen ist, dass sie für jedes Betriebssystem neu entwickelt oder migriert werden müssen. Hier bietet eine Webanwendung klare Vorteile durch eine vereinfachte Softwareentwicklung, da bei dieser Art von Anwendung nicht auf die systemspezifischen Gegebenheiten eingegangen werden muss. Außerdem ist die Kompatibilität zu einzelnen Architekturen und Sicherheitskonzepten ebenfalls höher als bei nativen Applikationen [12].

„Aus Sicht von Wartbarkeit, Entwickelbarkeit sowie den Kosten ist die Entwicklung von Apps auf Basis von HTML5 und CSS3 die wohl beste Alternative.“ [12], S. 61.

In erster Instanz wird der Prototyp für den mobilen Bereich eine native Anwendung sein. Dies hat mehrere Gründe: Zum einen sind noch nicht alle Anforderungen an das System bekannt oder in Betracht gezogen worden. Zum anderen ist noch nicht klar in welcher Form der Prototyp in späteren Versionen die Hardware der einzelnen Geräte verwendet. Schlussendlich ist es später immer noch möglich auf eine webbasierte Lösung zu wechseln. Hierbei können die gesammelten Erfahrungen mit den nativen sowie webbasierten Lösungen weiter in die Entwicklung des Systems mit einfließen.

4. SZENARIO

Um die einzelnen Funktionen des Portalsystems zu überprüfen wurde ein Testszenario entworfen. Hierbei handelt es sich um eine Chat-Anwendung, bei der es zusätzlich zum Senden von Textnachrichten möglich sein soll, Inhalte im DMS auf einfache Art und Weise für Chat-Teilnehmer freizugeben. Hat sich ein Anwender an das System angemeldet, ist es ihm möglich mit anderen Nutzern in Verbindung zu treten. Dabei sollen ihm nicht alle am System angemeldeten Benutzer angezeigt werden, sondern nur diejenigen, die er in seiner Freundesliste hat. Um dies zu ermöglichen ist es nötig zusätzlich eine Datenbank mit in das System einzubinden. In dieser Datenbank werden alle Informationen gespeichert, die den Chat sowie die Freundeslisten des einzelnen Anwenders betreffen. Darüber sollen später die Webanwendung sowie die mobile Anwendung miteinander kommunizieren.

Ziel dieses Szenario ist es, die Tauglichkeit der Referenzarchitektur sowie der Anwendungen und deren Kommunikation untereinander zu testen. Die

Ergebnisse fließen erneut in den Entwicklungsprozess des Projekts ein. Gegebenenfalls müssen die verwendeten Technologien angepasst oder ausgetauscht werden.

5. LITERATURVERZEICHNIS

- [1] Koch, Richter. 2007. Enterprise 2.0. Planung, Einführung und erfolgreicher Einsatz von Social Software in Unternehmen, Oldenbourg-Verlag.
- [2] WMC². 2011. WMC² Startseite. <http://wmc2.informatik-reutlingen.de/>.
- [3] Hochschule Reutlingen. 2011. Hochschule in Zahlen. <http://www.reutlingen-universi-ty.de/hochschule/informatives/portrait/hochschule-in-zahlen.html>.
- [4] Tiki – Wiki CMS Groupware. 2011. Tiki – Wiki CMS Groupware Startseite. <http://info.tiki.org/tiki-index.php>.
- [5] Liferay. 2011. Liferay Portal Wiki. <http://www.liferay.com/community/wiki/-/wiki/Main/Liferay+Portal>.
- [6] Fetzer, Höhl, Herbst, Polkehn. 2011. MyPaed, Meine persönliche Studienumgebung, TU Darmstadt. <http://www.mypaed.tu-darmstadt.de/>.
- [7] Hinderer, Henning; Gurzki, Thorsten; Vlachakis, Joannis; Kirchhof, Anja. 2006. Die Fraunhofer Portal Analyse und Design Methode (PADEM), Fraunhofer Institut für Arbeitswirtschaft und Organisation (IAO). http://www.e-bus-ness.iao.fraunhofer.de/Images/Whitepaper_PADEM_tcm462-31435.pdf.
- [8] SAML. 2011. SAML Startseite. <http://saml.xml.org/>.
- [9] Google. 2011. Google Web Toolkit. <http://code.google.com/webtoolkit/>.
- [10] Hanson, Robert; Tacy, Adam. 2007. GWT im Einsatz - AJAX-Anwendungen entwickeln mit dem Google Web Toolkit, Carl Hanser Verlag München Wien.
- [11] Google. 2011. GWT Mobile Toolkit. <http://code.google.com/p/gwt-mobile-webkit/>.
- [12] Oelmaier, Hörtreiter, Seitz. 2011. Apple's iPad im Enterprise-Einsatz, Springer-Verlag Berlin
Letzter Zugriff auf sämtliche Internetquellen, 31.03.2011

⁴⁰ UMTS – Universal Mobile Telecommunications System