

Entwicklung einer Software zur Präsentation von Multimediainhalten auf einer Videowand bei Sportveranstaltungen

Studiengang Medieninformatik der
Technischen Hochschule Mittelhessen

Bachelorarbeit

vorgelegt von

Daniel Beckmann

geb. in Hanau

durchgeführt bei CHIPZEIT.DE, Altstadt

Referent der Arbeit: Prof. Dr. Cornelius Malerczyk
Korreferentin der Arbeit: Dipl.-Math. (FH) Sabine Langkamm
Betreuer bei CHIPZEIT.DE: Dipl.-Ing. Ingo Morton



Fachbereiche

Informationstechnik-Elektrotechnik-Mechatronik IEM
und
Mathematik, Naturwissenschaften und Datenverarbeitung MND

Friedberg, 2011

Danksagung

An dieser Stelle möchte ich mich ganz herzlich bei den Personen bedanken, die zum Gelingen dieser Arbeit beigetragen haben. Zunächst bei meinen Betreuern Prof. Dr. Cornelius Malerczyk und Dipl.-Math. (FH) Sabine Langkamm, die mir die Möglichkeit gaben, meine Bachelorarbeit an der TH Mittelhessen durchführen zu können.

Diese Arbeit wurde außerdem im Unternehmen CHIPZEIT.DE in Altenstadt durchgeführt. Hier möchte ich Ingo Morton vielmals für seine große Unterstützung danken. Auch hinsichtlich der bereitgestellten Hardware, ohne die diese Arbeit gar nicht möglich gewesen wäre.

Dank gebührt außerdem Laura Unger und Andreas Vetterlein für die unschätzbare Geduld und gewissenhafte Fehlerkorrektur.

Schließlich möchte ich mich noch bei meiner Familie bedanken, die mich während meines gesamten Studiums in jeder Hinsicht unterstützt und somit zum erfolgreichen Abschluss dieser Arbeit maßgeblich beigetragen hat.

Selbstständigkeitserklärung

Ich erkläre, dass ich die eingereichte Bachelorarbeit selbstständig und ohne fremde Hilfe verfasst, andere als die von mir angegebenen Quellen und Hilfsmittel nicht benutzt und die den benutzten Werken wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Bruchköbel, August 2011

Daniel Beckmann

Inhaltsverzeichnis

Danksagung	i
Selbstständigkeitserklärung	iii
Inhaltsverzeichnis	v
Abbildungsverzeichnis	vii
1 Einleitung	1
1.1 Motivation	1
1.2 Problemstellung	2
1.3 Zielsetzung	5
1.4 Organisation der Arbeit	6
1.5 Zusammenfassung der wichtigsten Ergebnisse	6
2 Stand der Forschung	9
2.1 Einleitung	9
2.2 Begriffsdefinitionen	10
2.2.1 Echtzeitbedienung	10
2.2.2 Usability	10
2.3 Mensch-Computer-Interaktion und Echtzeit	11
2.4 Zusammenfassung	15
3 Videowände und Software	17
3.1 Einleitung	17
3.2 Software	18
3.2.1 Präsentations-Software	18
3.2.2 Großanzeigen-Software	19
3.2.3 Playout-Systeme	21
3.2.4 Sportspezifische Software	22
3.3 Zusammenfassung	23
4 Live-Video Systeme	25
4.1 Videosysteme	25

4.1.1	Videokameras und Camcorder	26
4.1.2	Netzwerk-Kameras	26
4.2	Videomischer	27
4.3	Videowände	28
4.4	Zusammenfassung	30
5	Grundlagen der Programmierung	31
5.1	Einleitung	31
5.2	Flash/Flex	32
5.3	GreenSock Tweening Platform	34
5.4	Actionscript und MXML	34
5.5	Zusammenfassung	35
6	Konzeptentwicklung	37
6.1	Einleitung	37
6.2	Analyse der Anforderungen	38
6.3	Benutzerseitige Konzeptentwicklung	40
6.3.1	Use Cases	44
6.4	Zusammenfassung	45
7	Prototypische Realisierung	47
7.1	Technische Konzepte	47
7.1.1	Bedienfehler bei der Echtzeitbedienung	48
7.1.2	Allgemeine Konzepte	48
7.2	Benutzeroberfläche	49
7.3	Architektur	55
7.4	Zusammenfassung	59
8	Ergebnisse	61
8.1	Entwickelte Konzepte	61
8.2	Betrachtung des Prototypen	62
8.3	Evaluation	63
8.3.1	Versuchsaufgaben	63
8.3.2	Ergebnisse der Evaluation	64
8.4	Zusammenfassung	66
9	Zusammenfassung und Ausblick	67
A	Inhalte der CD	69
B	Actionscript 3 und MXML	71
	Glossar	74
	Literaturverzeichnis	77

Abbildungsverzeichnis

1.1	Videowände	1
1.2	Videomischer	3
1.3	Präsentations Software	4
1.4	Ergebnisanzeigen im Sport	5
2.1	Usability does Matter	10
2.2	Handgestensteuerung	11
2.3	Komponenten des Mensch-Maschine-System	12
2.4	Bildverarbeitungsmethoden	13
2.5	Inhaltsbasiertes Video-Streaming	14
3.1	Großanzeigen Cluster	20
3.2	Playout-System	21
3.3	Stadion-TV	23
4.1	Videokamera und Camcorder	25
4.2	In-/Externe Videokarte	26
4.3	Netzwerk-Kameras	27
4.4	Videomischer Aufbau	28
4.5	Videowände	29
4.6	Schematischer Aufbau PC und Videowand	29
5.1	Flash Flex und Air-Anwendung	32
5.2	Benchmark Java und AIR	33
5.3	AIR Anwendungen	35
6.1	Bedienverzögerungen	39
6.2	Skizze Videomischer-Software	42
6.3	Use Case Diagramm #1	44
6.4	Use Case Diagramm #2	45
7.1	Bus Design	50
7.2	Fadebar Skin	51
7.3	Effekte	52
7.4	Vorschau-Element	52

ABBILDUNGSVERZEICHNIS

7.5	Oberflächendesign	54
7.6	Interface Diagramm Medienobjekt	55
7.7	Klassendiagramm	57
7.8	Bühnenschichten	58
8.1	Vergleich Prototyp und Playout-System	62
8.2	Ergebnis der Benutzerbefragung	64

Kapitel 1

Einleitung

1.1 Motivation

Die Nutzung von Videowänden auf öffentlichen Veranstaltungen nimmt von Jahr zu Jahr zu. Immer häufiger werden die Wände auch im kleineren Rahmen eingesetzt und bleiben nicht ausschließlich den größeren Veranstaltungen, wie dem Public Viewing oder Konzerten vorbehalten. Allein in Deutschland gibt es inzwischen über 50 Anbieter für Videowände und Anzeigetafeln [Par11]. Der Standard sind heutzutage LED-Wände, da sie durch ihre geringe Stromaufnahme, hohe Effizienz und lange Lebensdauer im Vergleich am günstigsten sind und zudem gute Ergebnisse liefern. Auf den Videowänden können Bild- oder Videoinhalte von verschiedenen Quellen angezeigt werden. Eines der beliebtesten Einsatzgebiete ist hier die Werbung. Diese ist auf einer Videowand gut sichtbar und kann als Bild, Animation oder Werbefilm prominent präsentiert werden. Früher war man auf einfache Mittel, wie Werbebanner beschränkt, doch durch diese Technik ergeben sich viel mehr Möglichkeiten. Als Vermarktungskonzept sind Videowände für Veranstalter also ideal.

Ein heute häufig genutzter Einsatzort von Videowänden sind Sportveranstaltungen. Hier können vielseitige Inhalte rund um die Veranstaltung gezeigt werden. Neben Sponsoranzei-



Abbildung 1.1: Videowand in Nahaufnahme (Quelle: SEP Inc.) und im Einsatz auf einer Laufveranstaltung (Quelle: Norbert Wilhelmi).

gen sind hier auch Live-Videos von der Veranstaltung oder Ergebnisse üblich. Den Sportveranstaltungen an sich ist nach wie vor eine sehr hohe Bedeutung zuzurechnen. Nach einer Studie gaben im Jahr 2000 über 30 Millionen der Deutschen an, jedes Jahr Sportveranstaltungen zu besuchen [Wil05]. Die Zahl der Sportveranstaltungen allein in der Schweiz beträgt jährlich zirka 230.000 [BLMS08]. Die Kombination von Videowänden und Sportveranstaltungen ist also durchaus interessant zu betrachten.

Über die Kombination von beidem lassen sich folgende Einsatzgebiete feststellen:

- Werbung
- Live Übertragungen von Kameras, Fernsehübertragungen
- Bilder der Veranstaltung
- Videoeinspielungen
- Informationen rund um die Veranstaltung
- Spielstand-/Ergebnisanzeige

In dieser Arbeit soll sich auf diese Punkte beschränkt werden, weil damit die wichtigsten Einsatzgebiete abgedeckt sind.

1.2 Problemstellung

Um Inhalte auf einer Videowand darzustellen, müssen Daten aus einer externen Quelle eingespeist werden. Dazu gibt es zwei Möglichkeiten, die am häufigsten verwendet werden:

- Anschluss an einen Videomischer
- Anschluss an den Computer

Der Weg über den Videomischer stellt hier die High-End-Variante dar, die auch von den größten Anwendern, wie beispielsweise der Bundesliga verwendet wird [PHPP11b]. An einem Videomischer können sehr viele unterschiedliche Eingabequellen genutzt werden, zwischen denen in Echtzeit beliebig hin- und hergeschaltet werden kann. An die meisten Videomischer kann auch der Computer als Eingabequelle hinzugeschaltet werden. Somit ist diese Lösung technisch gesehen die bessere — da sie die zweite Möglichkeit mit beinhaltet. Es gibt hier auch die Möglichkeit zwischen verschiedenen Übergängen (Transitions) zu wählen, Überlagerungen (Overlays) anzuzeigen und viele unterschiedliche Effekte einzusetzen. Diese Variante ist jedoch mit einem hohen Aufwand verbunden, denn es wird zusätzliche Hardware benötigt und es erfordert eine zusätzliche Regietechnik.

Die andere Möglichkeit ist die Nutzung des Computers als Eingabemedium. Hier müssen bisher im Vergleich zu Videomischern aber noch viele Einbußen gemacht werden. Denn

hier werden häufig Präsentationsprogramme, wie Powerpoint¹, oder auch Programme der Videowandhersteller, wie das LED Studio von Linsn², eingesetzt. Diese sind in ihrem Funktionsumfang jedoch sehr eingeschränkt, vor allen Dingen, wenn es um Steuerbarkeit geht. Wenn eine Präsentation läuft, können keine direkten Interaktionen mehr mit dem System gemacht werden. Eine Steuerung der Inhalte in Echtzeit ist somit nicht möglich. Besonders bei Sportveranstaltungen ist das schnelle gezielte Wechseln zu bestimmten Inhalten aber ausschlaggebend. Wenn bei einem Marathonlauf die ersten Läufer ins Ziel kommen, sollte hier etwa direkt auf ein Live-Kamerabild gewechselt werden können. Oder es müssen im laufenden Betrieb noch neue Werbeeinblendungen zugeschaltet werden.

Hier wird ersichtlich, dass diese Programme für den Sportbereich nicht vorgesehen sind. Diese großen Beschränkungen werden meist so hingenommen, obwohl so viel Potenzial hinter Videowänden steckt. Im Vergleich zu Videomischern sind hier also recht wenig Freiheiten gegeben. Man wird durch die Software beschränkt. Jedoch hat der Computer als Eingabequelle sicherlich großes Potenzial, einen Videomischer zu ersetzen.

Aus einer Publikation von Stadionwelt geht hervor, dass auch häufig sogenannte Playout-Systeme verwendet werden, die hauptsächlich für die Anzeige von (Live-)Videoinhalten konzipiert sind [PHPP11b]. Einige dieser Playout-Systeme setzen die Konzepte eines Videomischers eins zu eins im Computer um und bilden diesen in der Software nach. Jedoch werden in erster Linie nur externe Videoquellen für die Anzeige unterstützt. Daneben lassen sich zwar auch Bilder und Videoclips einspielen, jedoch nur sehr eingeschränkt. Diese sind eher dazu gedacht, Informationen auf die Anzeige zu bringen, oder kurze Werbeclips auszustrahlen. Die Bilder und Videos lassen sich untereinander jedoch nicht mischen, denn diese Funktion ist nur für die eingebundenen Kamerabilder vorgesehen. Der Einsatzort solcher Anwendungen sind daher eher Konzerte und Live-Auftritte, bei denen mehrere Kameras gleichzeitig verwendet werden und die jeweilige Anzeige flexibel gewechselt werden soll. Daher können diese

¹<http://www.microsoft.com>

²<http://www.linsn.com>



Abbildung 1.2: Videomischer der Firma Edirol mit und ohne Bildschirm (Quelle: Roland Corporation).

1. EINLEITUNG

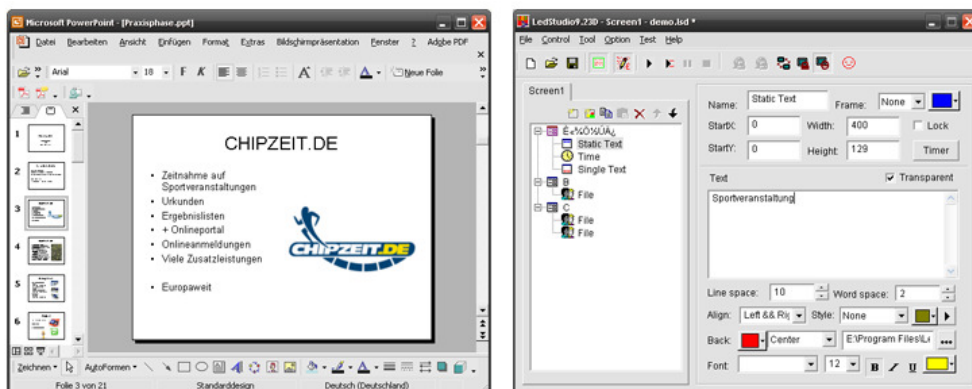


Abbildung 1.3: Powerpoint oder LED Studio bieten keine Möglichkeit der Echtzeitsteuerung von Präsentationen an (Quellen: Microsoft Powerpoint und Linsn LED Studio).

Systeme im Sportbereich nur in Stadien effektiv eingesetzt werden, wenn verschiedene Live-Kamerabilder auf eine Großanzeige übertragen werden sollen. Durch diesen Einsatzort wird auch die Zielgruppe dieser Programme recht schnell klar — nämlich Medientechniker, oder anderes technisches Personal. Das wird auch durch die Benutzeroberfläche der Programme ersichtlich. Es werden sehr viele Funktionen angeboten, welche die Benutzeroberfläche recht komplex machen und nur dem Experten eine effektive und effiziente Bedienung anbieten. Ein Schwerpunkt auf die Optimierung der Mensch-Computer-Schnittstelle wurde hier demnach nicht gelegt.

Wie bereits erwähnt, wird der sportliche Aspekt von dieser Software auch nicht berücksichtigt. Die Anzeigemöglichkeiten von Videowänden bei Sportveranstaltungen werden deshalb gar nicht ausgereizt, denn die Anzeige von Spielständen oder Ergebnissen wird nicht unterstützt.

Viele Alternativen existieren softwareseitig nicht. Es existieren noch sportartspezifische Programme, die bevorzugt in Stadien beim Fußball eingesetzt werden. Hier lassen sich viele Anzeigen rund um die Sportart einrichten, wie Informationen zu Spieler-Auswechslungen, Verwarnungen oder Punkteständen. Diese Programme sind für die jeweilige Sportart ideal, aber daher auch zu spezialisiert und für die allgemeine Präsentation von Multimedia-Inhalten nicht vorgesehen.

Hier wird das Problem ersichtlich, denn zum Einsatz einer Videowand auf Sportveranstaltungen unter Nutzung des Computers als Eingabemedium existieren bisher keine wissenschaftlichen Untersuchungen, die Konzepte für die Steuerung von multimedialen Inhalten in Echtzeit anbieten. Auch ist keine Software vorhanden, die in allen Sportbereichen gleichwertig für diesen Zweck eingesetzt werden kann. Powerpoint und ähnliche Programme reichen für den Einsatz in diesem Gebiet oft nicht aus, werden aber bevorzugt verwendet, weil sie sehr bekannt und einfach einzusetzen sind. Auf der anderen Seite sind viele Programme auch zu spezifisch auf eine Sportart ausgelegt und bieten keine allgemeinen Funktionalitäten.



Abbildung 1.4: Ergebnisanzeige bei Laufveranstaltungen und beim Fußball (Quellen: Mac Pale, Bundesliga).

1.3 Zielsetzung

Wie in der Problemstellung erwähnt, stellen Sportveranstaltungen ganz besondere Anforderungen an eine Software, die einen Videomischer ersetzen soll. Das Ziel dieser Arbeit besteht deshalb darin, die Anforderungen von Sportveranstaltungen im Bezug auf Videowände zu analysieren und aus den daraus gewonnenen Erkenntnissen gezielt Konzepte zu entwickeln. Diese sollen dann prototypisch in einer Software umgesetzt und danach evaluiert werden. Hierbei sollen insbesondere die Anforderungen an die Software als Mensch-Maschine-Schnittstelle unter Berücksichtigung von Effektivität, Effizienz und Anwenderzufriedenheit beachtet werden. Diese Betrachtungen sind insbesondere wichtig, da ein solches System in Echtzeit bedient werden muss.

Die Anforderungen die sich an eine Videowand Software im Bereich Sportveranstaltungen stellen und zusätzlich mit einbezogen werden sollen, sind unter anderem folgende:

1. Es muss schnell und gezielt agiert werden können, schnelle Wechsel auf gewünschte Mediendateien oder Live-Videos sollen deshalb möglich sein. Diese Wechsel sollen automatisch oder manuell gesteuert werden können.
2. Auch in Stresssituationen sollten keine Bedienfehler auftreten. Beispielsweise sollten nicht aus Versehen andere Bedienaktionen außerhalb des Programms auf der Videowand auftauchen, sondern nur die Inhalte, die dafür vorgesehen sind.
3. Auch im Betrieb sollten sich Medien beliebig hinzufügen oder auch aus der Anzeigeliste entfernen lassen.
4. Die Software soll auch ohne Fachkenntnis bedienbar sein, sprich: Sie soll schnell erlernbar und übersichtlich sein.
5. Auf der Videowand sollen Live-Ergebnisse dargestellt werden können, diese sollen sich von einem externen Datenanbieter abrufen und in das System einbinden lassen (siehe Abbildung 1.4).

Besonders durch den letzten Punkt ergeben sich im Sportbereich sehr viele Möglichkeiten. Deshalb soll sich in dieser Arbeit auf Rad- und Laufveranstaltungen beschränkt werden. Demnach ist hier die Zielsetzung, ein Modul auf Basis von „Top Ten“-Ergebnissen zu implementieren.

In erster Linie soll das System für die LED-Videowand „DMT Pixelscreen“, unter Nutzung von Live-Videos von dem Kamerasystem „AXIS P5534-E“ konzipiert werden. Jedoch soll es sich auch problemlos mit anderen Videowänden und Kamerasystemen nutzen lassen.

1.4 Organisation der Arbeit

In den nächsten zwei Kapiteln wird genauer auf den Stand der Forschung und Technik eingegangen. Zuerst werden aktuelle Untersuchungen in den verschiedenen Bereichen der Mensch-Computer-Interaktion und Echtzeitbedienung von Multimediasystemen betrachtet. Daraufhin werden die heute gängigen Verfahren zum Bespielen von Videowänden mit Multimediainhalten in Bezug auf die Software betrachtet. Hier werden die verschiedenen Softwaretypen näher betrachtet und nahe gelegt, warum die in der Einleitung genannten Probleme nicht durch diese Software gelöst werden.

Die zwei darauf folgenden Kapitel beschäftigen sich mit den Grundlagen. Erst in Bezug auf Live Video Systeme. Hier wird näher auf Videosysteme, Videomischer und Videowände eingegangen. Danach folgen die Grundlagen der Programmierung, die für die Entwicklung des Prototyps ausschlaggebend sind.

In den Hauptkapiteln werden die Anforderungen von Sportveranstaltungen aus verschiedenen Blickwinkeln betrachtet und benutzerseitige Konzepte entwickelt. Hierzu werden die späteren Nutzergruppen identifiziert und Use Cases (Anwendungsfälle) herausgestellt. Nach diesen Untersuchungen wird auf die technischen Konzepte eingegangen, also wie die benutzerseitigen Konzepte in einer Software umgesetzt werden können. Die Konzepte eines Videomischers bieten hier die Grundlagen für die Bedienkonzepte, da dieser eine hardwareseitige Echtzeitbedienung bereits möglich macht. Hierzu wird einerseits die Benutzeroberfläche und andererseits die Architektur entwickelt und näher besprochen.

In den letzten Kapiteln wird auf die erzielten Ergebnisse eingegangen. Zusätzlich werden die Ergebnisse der Evaluation des Prototypen angeführt und daraufhin näher betrachtet. Darauf folgt eine Zusammenfassung dieser Arbeit und ein Ausblick.

1.5 Zusammenfassung der wichtigsten Ergebnisse

Die Betrachtungen dieser Arbeit zeigen, wie die Präsentation von Multimediainhalten auf einer Videowand möglich gemacht werden kann. Aus den Betrachtungen von Sportveranstaltungen wurde ersichtlich, dass eine Bedienung der Software in Echtzeit einen sehr wichtigen Faktor darstellt. Somit wurden unter dem Vorbild des Videomischers, benutzerseitige und technische Konzepte entwickelt, die eine Präsentation dieser Inhalte in Echtzeit

möglich machen. Diese Konzepte wurden im nächsten Schritt prototypisch in einer Flash-Anwendung umgesetzt. Hierbei stellte sich auch heraus, dass die Benutzeroberfläche die wichtigste Grundlage für die Echtzeitbedienung darstellt.

Aus diesen Betrachtungen heraus wurden verschiedene Bedienelemente auf der Benutzeroberfläche umgesetzt. In einer Tabelle werden die einzelnen Medienobjekte, die präsentiert werden sollen, untereinander angezeigt. Auf Knopfdruck können diese auf die Videowand geschaltet werden. Hierzu gibt es einen Regler, über den manuelle Übergänge erzeugt werden können. Andererseits können die Objekte auch nacheinander über einen automatischen Modus angezeigt werden. Zum Übergang stehen hier verschiedene Effekte bereit, die in Echtzeit angewendet werden können.

Als weitere Unterstützung in der Bedienung wurden zwei Vorschaufenster entwickelt. In dem Hauptfenster sieht der Benutzer die aktuelle Vorschau der Videowand-Ausgabe. Diese ist nützlich, falls der Bedienende keinen direkten Blick auf die Wand haben sollte. Das andere Fenster zeigt eine Vorschau des Objekts, das als nächstes angezeigt werden soll. Diese Bedienelemente stellen die wichtigsten Komponenten der Benutzeroberfläche dar.

Eine effektive und effiziente Bedienung in Echtzeit wird durch den Prototypen gewährleistet, da die Präsentationsebene von der Benutzeroberfläche getrennt ist. Bei der Benutzeroberfläche wurde auf unnötige Bedienelemente verzichtet und somit eine gute Übersichtlichkeit geschaffen. Der Benutzer hat jederzeit die volle Kontrolle über die Inhalte auf der Videowand und kann selbst bestimmen, wann welches Medienobjekt angezeigt werden soll.

Die Ergebnisse und die Evaluation des Prototypen haben gezeigt, dass die Zielsetzung nach Effektivität, Effizienz und Anwenderzufriedenheit zufriedenstellend erfüllt worden sind.

Kapitel 2

Stand der Forschung

Dieses Kapitel gibt einen Überblick über den aktuellen Stand der Forschung auf dem Gebiet der Mensch-Computer-Interaktion, unter dem Aspekt der Echtzeit. Hierbei sollen insbesondere Anwendungen dieses Fachgebiets im Bereich Multimedia betrachtet werden. Hierzu werden die wichtigsten in der Literatur und Publikationen beschriebenen Ansätze zu dem Thema vorgestellt.

2.1 Einleitung

Interaktive Systeme werden in der heutigen Zeit zunehmend wichtiger. Vor allem im Bereich der Multimedia-Anwendungen. Als Multimedia-Anwendungen werden hauptsächlich Desktop-Anwendungen bezeichnet, die „digitale Medien integrieren, wie beispielsweise Text, Pixelbilder, Grafik, Video oder Ton“ [Iss02]. Andererseits aber auch dementsprechende Anwendungen im Internet. Dadurch ergeben sich viele verschiedene Anwendungsgebiete. Mit zu den beliebtesten Diensten im Internet gehören Videoportale wie YouTube ¹ oder MyVideo ². Hier ist das Internet im Sportbereich sicherlich auch mit dem (Live-)Streaming von Sportereignissen an eine bedeutsame Stelle gerückt. Über das Streaming können die Videodaten von beliebig vielen Nutzern abgerufen werden.

Echtzeit spielt für diese Systeme auch in vielerlei Hinsicht eine entscheidende Rolle. Auf der einen Seite bei den Serviceanbietern, bei denen immer mehr Ressourcen benötigt werden, um die Inhalte in Echtzeit zu vielen Endpunkten in hohen Auflösungen übertragen zu können. Auf der anderen Seite beim Anwender, weil der Benutzer ein schnelles Verhalten auf seine Bedienaktionen in der Abspieldsoftware erwartet. Da interaktive Systeme immer wichtiger werden sind die Folgen, dass die Bedeutung der Mensch-Computer-Interaktion auch deutlich zunimmt [HW09]. Aber auch die Bedeutung von Echtzeit und die Forderung nach einer guten Usability. Im multimedialen Anwendungsbereich nimmt auch der so genannte „Joy-of-Use“ [Hat00, Ree04] eine wichtige Position ein. Der „Joy-of-Use“ steht hier für den Nutzerspaß, den der Anwender beim Benutzen eines Systems haben kann. Besonders im

¹<http://www.youtube.de>

²<http://www.myvideo.de>

Bereich der Medien ist der Benutzerspaß enorm wichtig, denn beim Benutzen werden Spaß und Unterhaltung vorausgesetzt.

2.2 Begriffsdefinitionen

2.2.1 Echtzeitbedienung

Der Begriff Echtzeit wird oft fälschlicherweise in Bezug auf die Geschwindigkeit eines Systems verwendet. Er sagt jedoch lediglich aus, dass eine Anwendung innerhalb eines vorgegebenen Zeitrahmens reagieren muss. In Bezug auf die Bedienung also, dass auf alle Bedienaktionen innerhalb einer vorgegebenen Zeit die gewünschte Reaktion ausgeführt werden muss. Dieser Zeitrahmen muss bei Anwendungen im Multimedia Bereich sehr kurz sein, um die gewünschten Operationen und das menschliche Empfinden nicht zu beeinträchtigen. Im Falle einer Präsentationssoftware bedeutet Echtzeitbedienung, dass sich die angezeigten Inhalte jederzeit ändern lassen und dass Bedienaktionen innerhalb einer unmittelbar kurzen Zeit von der Software umgesetzt werden und sichtbar sind.

2.2.2 Usability

Unter Usability wird die Gebrauchstauglichkeit eines Produkts verstanden. Diese ist definiert durch die ISO Norm 9241 [92498], als „das Ausmaß, in dem ein Produkt durch bestimmte Benutzer in einem bestimmten Nutzungskontext genutzt werden kann, um bestimmte Ziele effektiv, effizient und zufriedenstellend zu erreichen“. Auf die drei genannten Hauptziele der Gebrauchstauglichkeit wird im Folgenden näher eingegangen:

- Effektivität - Die vorliegenden Aufgaben lassen sich mit einem System vollständig und korrekt erfüllen.
- Effizienz - Die vorliegenden Aufgaben lassen sich zuverlässig und mit möglichst wenig Aufwand lösen. Die Effizienz steht im Verhältnis zur Effektivität und der Zeit, die der Benutzer zur Erledigung seiner Aufgaben benötigt.
- Zufriedenheit - Beschreibt die Zufriedenheit der Benutzer bei der Nutzung eines Systems.



Abbildung 2.1: Wenn Usability fehlschlägt (Quelle: www.gruenderszene.de).

Aufgabenangemessenheit	Das System unterstützt den Benutzer bei der effektiven und effizienten Erledigung seiner Aufgaben
Selbstbeschreibungsfähigkeit	Das System enthält Erläuterungen und ist ausreichend verständlich
Steuerbarkeit	Der Benutzer soll in der Lage sein, den Dialogablauf zu beeinflussen
Erwartungskonformität	Erwartungen, Eigenschaften und Gewohnheiten der Benutzer werden unterstützt
Fehlertoleranz	Fehler können mit keinem oder geringen Korrekturaufwand behoben werden
Individualisierbarkeit	Der Benutzer kann das System an seine individuellen Bedürfnisse anpassen
Lernförderlichkeit	Der Benutzer kann das System mit geringem Aufwand erlernen und wird durch es unterstützt

Tabelle 2.1: Die sieben Grundsätze der Dialoggestaltung nach DIN EN ISO 9241-110

Um Usability in einer Software zu erreichen, gibt es eine unmittelbar große Menge an Richtlinien, die durch Usability Guidelines festgelegt werden. Die sehr häufig genannten Richtlinien in Bezug auf eine Norm, sind die sieben Grundsätze der Dialoggestaltung [92406]. Diese sind in obiger Tabelle abgebildet.

2.3 Mensch-Computer-Interaktion und Echtzeit

Im Bereich der Echtzeitbedienung von Software beschäftigen sich wissenschaftliche Untersuchungen oft mit dem Thema der multimodalen Interaktion. Bei dieser Art der Interaktion werden mehrere Modalitäten (Sinne) zur Bedienung verwendet und es können beispielsweise Sprache oder Gesten, aber auch klassisch Tastatur und Maus zum Einsatz kommen. Ein heute beliebtes Gebiet ist die Handgestenerkennung, sprich die Steuerung des Computers mit

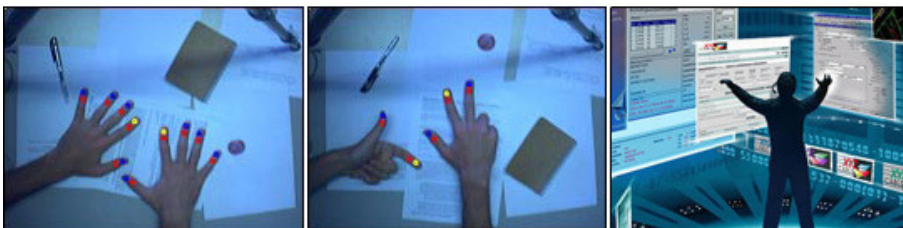


Abbildung 2.2: Durch die Finger-Klassifikation kann mit Computerprogrammen interagiert werden (Quellen: [HB01], www.mvsrp.de).

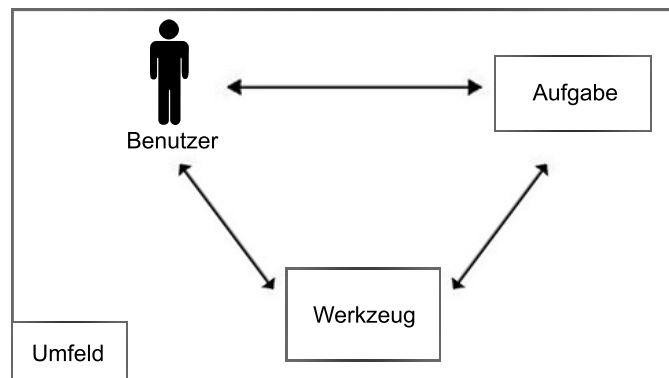


Abbildung 2.3: Die vier prinzipiellen Komponenten eines Mensch-Maschine-Systems. Der Benutzer will eine Aufgabe mit einem bestimmten Werkzeug in einem bestimmten Anwendungskontext erledigen.

der Hand ohne zusätzliche Geräte und ohne Kontakt zum Computer. Hier werden die Hände über eine oder mehrere Kameras gefilmt und die Bilder gleichzeitig zum Computer übertragen und ausgewertet. In den Untersuchungen werden Algorithmen und Verfahren entwickelt, um Finger- und Handgesten in Echtzeit erkennen und auswerten zu können [PJ00, HB01]. Mit als Haupt-Einsatzgebiet für die Steuerung des Computers über Handgesten werden hier Präsentationen genannt, denn der Benutzer kann über Fingergesten einfach vor und zurück schalten [HB01].

Diese Untersuchungen können jedoch für diese Arbeit nicht viel beitragen. Die Steuerung einer Anwendung über Handgesten, die später in Echtzeit arbeiten soll, ist zu fehleranfällig. Ein Multimedia System im Sportbereich wird auch weitaus mehr Steuerungsmöglichkeiten benötigen als „vor“ und „zurück“. Verwechselt das System verschiedene Gesten oder wird ein falsches Bedienelement gedrückt, dann wird ein Befehl falsch umgesetzt und schaltet etwa ein Medium weiter, anstatt zu pausieren. Dadurch kann eine fehlerfreie Bedienung nicht gewährleistet werden. Deshalb sollen ausschließlich Maus und Tastatur für eine präzise Steuerung verwendet werden.

Ansonsten gibt es viele Publikationen im Bereich der Mensch-Computer-Interaktion, die jedoch keine Betrachtung von Echtzeit mit einbeziehen. Jedoch können diese Untersuchungen für die Entwicklung von neuen Konzepten beitragen: In den beiden Standardwerken von Shneiderman [SP05] und Dahm [Dah05] beschäftigen sich die Autoren mit den Grundlagen der Mensch-Computer Interaktion und speziellen Aspekten von interaktiven Computersystemen. Es wird aufgezeigt, wie man die Mensch-Computer-Interaktion optimal gestalten kann und was im Design der Benutzeroberfläche zu beachten ist, um eine gute Usability zu gewährleisten. Das Ziel ist gebrauchstaugliche Software zu entwickeln und zu gestalten, damit der Nutzer sich schnell einarbeiten kann und Spaß bei der Benutzung hat. Shneiderman stellt in seinen Untersuchungen folgende Ziele der Anforderungsanalyse für Usability heraus

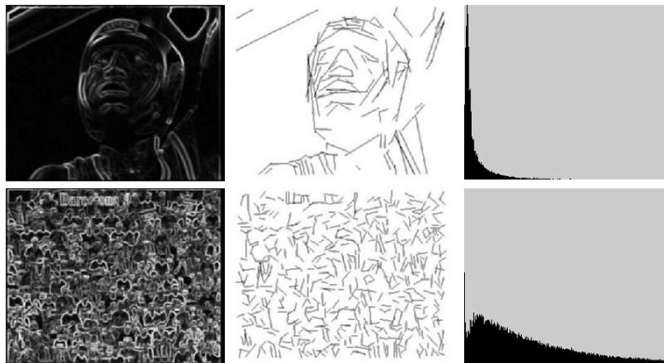


Abbildung 2.4: Kantenfilter und Segmentierung ermöglichen die Unterscheidung zwischen Spieler-Aufnahmen und Publikum (Quelle: [ABCB02]).

(Auszug) [SP05]:

- Die Nutzergruppe muss ermittelt werden und welche Aufgaben später mit dem System erledigt werden sollen. Die Aufgabenanalyse steht an einem zentralen Punkt, weil Oberflächen mit unzureichenden Funktionalitäten den Benutzer behindern und deshalb oft abgelehnt werden.
- Die Software muss zuverlässig sein und Aktionen müssen so funktionieren, wie vorgesehen. Ein einziges Ereignis mit irreführenden Daten oder unerwartetem Ergebnis reicht aus, um die Bereitwilligkeit des Benutzers zu senken, ein System auch weiterhin benutzen zu wollen.
- Der Designer sollte den Anwendungskontext der Benutzung bedenken und die dazugehörige Standardisierung, Integration, Konsistenz und Portabilität fördern.

Da die Usability eine große Wichtigkeit bei der Interaktion zwischen Mensch und Computer einnimmt, ist das so genannte Usability Engineering entstanden. Nach Richter und Flückiger „umfasst Usability Engineering Mittel und Techniken, um bei der Entwicklung neuer Software oder Produkte die angestrebte Usability zu erreichen“ [RF10]. Für die Entwicklung eines interaktiven Multimediasystems spielt Usability Engineering deshalb heute eine große Rolle. Man sieht dass vor allem bei einer Echtzeitbedienung Usability an einer wichtigen Stelle stehen muss. Es dürfen keine Bedienfehler auftreten und der Benutzer sollte sich gut zurechtfinden. Deshalb legen diese Untersuchungen einen Grundstein für diese Arbeit, auf den aufgebaut werden kann.

Betrachtet man Publikationen im Bereich von Multimedia und Sport, so findet man viele Untersuchungen, die Forderungen an Echtzeit stellen. In einer Studie von 2002 wurde untersucht, wie man Sportvideos automatisch mit Anmerkungen versehen kann, die auf den Inhalt rückschließen lassen [ABCB02]. Der Vorgang läuft automatisch während der Live-Aufnahme ab. Anmerkungen sollen hier Schlagwörter sein, wie die Sportart, oder Dinge, die unmittelbar mit dem Videoinhalt zu tun haben (spielende Mannschaften, Spielstände

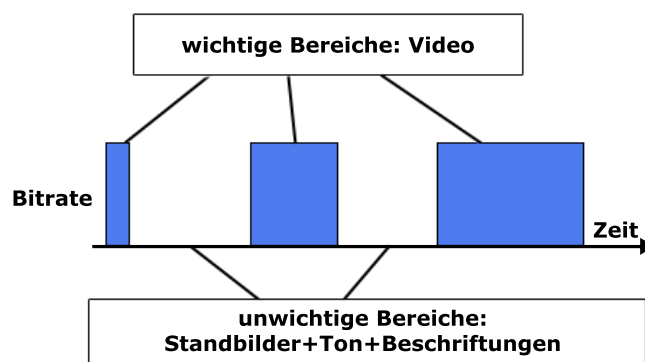


Abbildung 2.5: Nur wichtige Videoinhalte werden ausgestrahlt, unwichtige Bereiche werden als Standbilder mit Ton und Beschriftungen gestreamt (Quelle: [CZK01]).

oder auch zu welchem Zeitpunkt ein Punkt erzielt wurde). Das Video wird hier als zusammengesetztes Medium aus Einzelbildern, Grafiken, Text und Ton betrachtet. Zur Analyse wird jeder Bildausschnitt in seine einzelnen Teilelemente zerlegt und auf niedrigster logischer Ebene, durch die Kombination von verschiedenen Bildverarbeitungsmethoden (Kantenfilter, Segmentierung), untersucht (siehe Abbildung 2.4). Auf Basis dieser Untersuchungen kann dann auf den Bildinhalt geschlossen werden. Auch die unterschiedlichen Szenen werden logisch voneinander getrennt, wie etwa Interviews im Studio von Sportszenen. Diese Technik lässt sich im Softwarebereich sicherlich von sportartspezifischen Programmen nutzen, bietet aber keine Basis für Steuerung von Inhalten in Echtzeit. Hier werden lediglich Bildinhalte untersucht und automatisch annotiert.

Ein weiterer großer Bereich der Echtzeitanwendungen ist das Streaming von Multimedia-Inhalten. Durch die immer größer und beliebter werdenden Streaming- und Videoportale im Internet, werden immer mehr Ressourcen bei den Anbietern benötigt. Streaming steht für den Vorgang der Übertragung von Daten über Datenströme. Mit dem Videostreaming können Videos über das Netzwerk gesendet und dann bei den Empfängern abgespielt werden. Bekannte Dienste für Videostreaming sind Videoportale wie beispielsweise YouTube ³, oder Video-on-Demand Angebote (Maxdome ⁴ und Ähnliche).

Ähnlich der zuletzt genannten Inhaltsbasierten Annotationen von Sportvideos, ist eine Untersuchung der Columbia University, in der es um Inhaltsbasiertes Streaming [CZK01] geht. Allgemein ist in Netzwerken immer eine begrenzte Bandbreite vorhanden, die genutzt werden kann. Deshalb sollen Live Sportvideos vor ihrer Ausstrahlung in Echtzeit untersucht und nicht wichtige Inhalte ausgefiltert werden. Somit wird ein Streaming von ausschließlich bedeutsamen Inhalten erreicht und Bandbreite eingespart (siehe Abbildung 2.5). Unwichtige Inhalte werden durch Standbilder mit Ton und eventuellen Beschriftungen ersetzt, die im Vergleich relativ wenig Bandbreite benötigen. Dazu wurde ein Prototyp für die Sportarten

³<http://www.youtube.de>

⁴<http://www.maxdome.de>

Tennis und Basketball entwickelt. Wie auch in vorherigen Ansätzen wird durch eine Bildanalyse festgestellt, was sich in den Szenen abspielt. Unwichtige Inhalte, wie etwa Werbung oder Close-Up-Aufnahmen von Spielern und Fans, werden durch die Algorithmen ausgefiltert. Viele Untersuchungen beschäftigen sich auch direkt mit der Architektur der Hard- und Software auf Multimedia Streamingservern. Es wird ständig versucht die Architektur dieser Systeme zu verbessern und Prozesse zu optimieren, um genügend Leistung zur Verfügung stellen zu können. Laut einer Publikation der Universidad Autónoma de Madrid [VLAV94], „hängt die Erweiterung von Multimedia Netzwerken stark von der Unterstützung des Streamings der Mediendaten in Echtzeit ab“. Die Anforderungen in Bezug auf Echtzeit von solchen Systemen, werden somit oft von verschiedenen Seiten aus betrachtet und Probleme analysiert, um technische Lösungen für eine Architektur zu entwickeln [Nic90, VLAV94]. Folgende Aspekte sollen bei einem effizienten Management von Videostreams in Echtzeit eine Rolle spielen [VLAV94, Nic90]:

- Das System muss sowohl Audio- als auch Videodaten effizient auf der Festplatte verwalten und Zugriffe effizient planen, um die Anfragen von einem Echtzeitsystem zufriedenstellend bearbeiten zu können.
- In Videostreams müssen Bild und Ton synchron zueinander ablaufen.
- Interaktionen der vielen gleichzeitigen Benutzer an den Steuerelementen der Präsentationsebene mit Start/Stopp/Pause und Vorwärts-/Zurückspulen müssen effizient und effektiv bearbeitet werden.
- Die Bandbreite muss den Echtzeitanforderungen für den Transport der Daten über das Netzwerk genügen und der Dienstgüte entsprechen. Durch die immer höheren Ansprüche an Videoportale, wie Videos in hohen HD-Auflösungen und auch durch die Einführung des Digitalfernsehens über die Netzwerkleitung, ist immer mehr Bandbreite notwendig.

Um die Infrastruktur von solchen Streaming Servern verbessern zu können, müssen im ersten Schritt Evaluationen durchgeführt werden, um Probleme identifizieren zu können. Hierzu werden in einer Untersuchung [SCS⁺94] Verfahren entwickelt, um die Performanz von Streaming Services zu evaluieren und bewerten zu können.

Durch die vorangegangenen Untersuchungen wird deutlich, dass sich die Trends im Multimedia Bereich beim Videostreaming und Internet abspielen. Es werden neue Serverarchitekturen entwickelt und evaluiert, um den hohen Anforderungen, wie dem Abrufen von Videos in hohen Auflösungen, gerecht werden zu können. Usability spielt für diese Systeme auch eine zentrale Rolle.

2.4 Zusammenfassung

Aus den verschiedenen Betrachtungen wird ersichtlich, dass es kaum Veröffentlichungen im Bereich der Echtzeitbedienung von Multimedia-Anwendungen gibt. Es existieren viele Pu-

2. STAND DER FORSCHUNG

blikationen zur Mensch-Computer-Interaktion über Handgestenerkennung oder andere Eingabemodalitäten. Hier dreht sich die Frage um die Umsetzung der Gesten in Echtzeit und ohne große Verzögerungen. Ein anderer großer Bereich ist die Mensch-Computer-Interaktion an sich, jedoch ohne den Aspekt der Echtzeit. Hier werden viele Grundlagen aufgezeigt und geklärt, wie man die Interaktion zwischen Mensch und Computer optimieren kann. Ein anderer Forschungsbereich ist das Streaming von Multimedialinhalten. Der Fokus liegt hier im Streaming von Videoinhalten in Echtzeit und in hohen Auflösungen (HD), die heute schon fast als Standard vorausgesetzt werden.

Der aktuelle Stand der Forschung hilft bei der Lösung des gestellten Problems somit nicht direkt weiter. Lediglich die Erfahrungen aus den Grundsätzen der Mensch-Computer-Interaktion können die Untersuchungen dieser Arbeit im späteren Verlauf unterstützen. Somit ist eine genauere Untersuchung des Themas notwendig. Vor allem in Bezug auf Echtzeit und die sportlichen Anforderungen, die an das System gestellt werden.

Im nächsten Kapitel wird auf den Stand der Technik im Bereich der Software eingegangen.

Kapitel 3

Videowände und Software

Im letzten Kapitel wurde der Stand der Forschung im Bereich der Echtzeitbedienung von Multimedia-Anwendungen näher betrachtet. Die Betrachtungen haben gezeigt, dass fast keine wissenschaftlichen Untersuchungen zu diesem Bereich existieren. In diesem Kapitel werden einige Programme aus den vier mit Videowänden meist genutzten Software-Kategorien genannt und parallel dazu erläutert, inwiefern sie sich für den gewünschten Anwendungszweck der Echtzeitbedienung von Multimediainhalten verwenden lassen.

3.1 Einleitung

Videowände werden häufig über den Computer angesteuert, weil diese Variante den einfachsten und günstigsten Weg darstellt. Es werden häufig Präsentationsprogramme wie Powerpoint oder OpenOffice Impress verwendet, weil diese den meisten Anwendern bekannt sind. Im direkten Vergleich ist der Einsatz von Videomischern als Alternative mit zusätzlichem Aufwand und Kosten verbunden, die meistens vermieden werden sollen. Unbewusst oder bewusst für den Anwender, müssen dann aber noch sehr viele Einbußen in Funktionalität gemacht werden und die multimedialen Möglichkeiten einer Videowand werden gar nicht ausgereizt. Es wird auch schnell klar, dass diese Programme für den Sportbereich nicht vorgesehen sind.

Ein gutes Beispiel ist der Johannesbad Thermen Marathon in Bad Füssing ¹. Mit einer Teilnehmerzahl von knapp 2000 gehört die Veranstaltung schon zu den mittelgroßen Veranstaltungen ihrer Klasse. Auch bei dieser Veranstaltung kommt seit ein paar Jahren eine Videowand zum Einsatz. Diese wird auch über einen Laptop und Powerpoint mit Inhalten bespielt. Während der gesamten Veranstaltung wird die Videowand bisher nur für Sponsorwerbung und Werbung für die Johannesbad Therme verwendet. Nach Erfahrungen der Firma CHIPZEIT.DE ², die in der Zeitnahme auf Sportveranstaltungen tätig ist, kam der Wunsch des Veranstalters auf, auch Ergebnisse auf der Videowand zu veröffentlichen. Diesem Wunsch kann hier aber nur über zusätzliche Programme wie dem Webbrowser entgegen gekommen werden. Dazu muss die Präsentation erst beendet werden, um dann Inhalte aus

¹<http://www.thermen-marathon.de>

²<http://www.chipzeit.de>

einer externen Datenbank anzeigen zu können. Auch hier wäre der Einsatz eines Videomischers denkbar, der eine halb-transparente Überlagerung von Inhalten aus verschiedenen Datenquellen einrichten kann. Der Veranstalter als Anwender der Technik möchte hier aber auch keine zusätzlichen Kosten und einen höheren Aufwand haben. Eine Softwarelösung wäre hier demnach auch praktisch einzusetzen.

Von Fachpersonal werden häufig speziell für Videowände konzipierte Software oder Playout-Systeme verwendet. Diese Software bietet aber entweder in der Steuerbarkeit oder in der Flexibilität ebenfalls zu wenig Freiraum. Auf die genauen Probleme mit der bereits existierenden Software soll deshalb in diesem Kapitel genauer eingegangen werden.

3.2 Software

In diesem Teilkapitel werden vier Software-Kategorien betrachtet, die bisher am häufigsten mit Videowänden kombiniert werden. Die Kategorien werden kurz vorgestellt und danach die einzelnen Programme miteinander verglichen. Dazu wird überprüft, ob sich die Programme für eine Echtzeitbedienung unter sportlichem Aspekt nutzen lassen.

3.2.1 Präsentations-Software

Im Bereich der Präsentations-Software werden hier drei verschiedene Programme betrachtet und verglichen: Microsoft Powerpoint, OpenOffice Impress und Keynote von Apple zählen wohl zu den bekanntesten und meist genutzten Programmen im Präsentationsbereich. Grundsätzlich sind Präsentationsprogramme dazu gedacht Folien innerhalb eines Vortrags zu präsentieren. Diese einzelnen Folien können mit den Programmen bearbeitet werden und kombinieren verschiedene multimediale Inhalte wie Texte, Bilder, Grafiken, Diagramme, Tabellen, Videos oder Musik. Im Laufe einer Präsentation lassen sich dann Folie für Folie nacheinander anzeigen.

In Bezug auf Videowände und die Echtzeitbedienung sind Präsentationsprogramme in ihrer Funktion jedoch zu eingeschränkt. Es lassen sich zwar die meisten der gewünschten Multimedia Inhalte einfügen und auch automatisch nacheinander abspielen, jedoch ist die Software nicht flexibel genug. Die Folien laufen alle statisch nacheinander ab, ein direktes Navigieren zu einer Folie, in einem Mausklick und unsichtbar für den Betrachter, ist demnach nicht möglich.

Wie in Tabelle 3.1 zu erkennen ist, kommen alle genannten Programme in vielen Punkten nicht an die Anforderungen heran. Videostreams und externe Videoquellen, wie eine Webcam oder eine Videokarte, können erst gar nicht genutzt werden. Diese sind aber im Vergleich zu Videomischern schon ein Pflichtkriterium für die Software. Diese müssen unterstützt werden, denn auf Veranstaltungen werden Live-Videos häufig gefordert.

In Sachen Echtzeitbedienung ist in keiner Software eine flexible Bedienoberfläche vorgesehen. Präsentationsprogramme bieten hauptsächlich eine Steuerung für „vor“ und „zurück“. Über das Kontextmenü kann während einer Präsentation zu anderen Folien gewechselt werden. Bei dieser Methode werden die Bedienaktionen aber für den Betrachter sichtbar. Lediglich Keynote bietet die Möglichkeit eines sogenannten Moderationsmonitors an. Über diese

	Powerpoint	Impress	Keynote
Bilder	X	X	X
Text/Rich-Text	X	X	X
Videos	X	X	X
Externe Videoquellen	-	-	-
Videostreams	-	-	-
Flash-Inhalte	(X)	X	-
Anbindung an externe Datenquellen	-	-	-
Automatischer Folienwechsel	X	X	X
Manueller Folienwechsel	X	X	X
Animationen bei Übergang	X	X	X
Anzeigebereich anpassbar (Größe und Position)	-	-	-
Inhalte live einfügen	-	-	-
Bedienung in Echtzeit	-	-	((X))

Tabelle 3.1: Vergleich der einzelnen Präsentationsprogramme

Funktion sieht der Bedienende eine Vorschau der nächsten Folie und hat hier auch die Möglichkeit, eine Folie direkt anzusteuern — ohne das der Betrachter etwas davon mitbekommt. Trotzdem wird ersichtlich, dass Präsentationsprogramme nicht für die Anzeige von Inhalten auf einer Videowand bei Sportveranstaltungen vorgesehen sind. Es werden zu wenig Medientypen unterstützt und die Bedienung ist zu eingeschränkt.

3.2.2 Großanzeigen-Software

Die Software aus diesem Bereich ist speziell für Großanzeigen und Videowände gedacht und deshalb auch wesentlich besser geeignet. Das Programm LED Studio von Linsn³ ist ein sehr verbreitetes Programm, da es mit einem der Marktführer der LED-Steuerungskarten vertrieben wird. Nachteil ist jedoch, dass dieses Programm nur mit den Steuerungskarten von Linsn zusammenarbeitet. Somit kann es nicht in Kombination mit jeder Videowand verwendet werden. Der Multimedia Displayer⁴ hingegen ist ein hardwareunabhängiges Programm, dass mit jeder Videowand oder Anzeige zusammenarbeiten kann.

Der Fokus bei den genannten Programmen liegt bei der Unterstützung von sehr vielen verschiedenen Medienformaten und dem Ansteuern von mehreren Anzeigen parallel.

Viele Anforderungen in Bezug auf Medientypen werden durch diese Software abgedeckt. Auch kann man den Anzeigebereich und dessen Auflösung beliebig einstellen. Ein großer Nachteil ist hier jedoch, dass die Präsentation der Inhalte ebenfalls statisch abläuft. Gezieltes Ansteuern beliebiger Inhalte ist auch nur schwer möglich. Beim Multimedia Displayer können einzelne Inhalte lediglich mit Tastenkürzeln versehen werden über die sie dann jederzeit erreicht werden können. Auch ein manueller Folienwechsel lässt sich nur über diese Methode vollziehen. Vorteil des Multimedia Displayers ist aber, dass sich Inhalte jederzeit hinzufügen

³<http://www.linsn.com>

⁴<http://www.mediadisplayer.com/>

3. VIDEOWÄNDE UND SOFTWARE

	LED Studio	Multimedia Displayer
Bilder	X	X
Text/Rich-Text	X	X
Videos	X	X
Externe Videoquellen	X	X
Videostreams	-	-
Flash-Inhalte	X	-
Anbindung an externe Datenquellen	X	X
Automatischer Medienwechsel	X	X
Manueller Medienwechsel	-	(X)
Animationen bei Übergang	X	X
Anzeigebereich anpassbar (Größe und Position)	X	X
Inhalte live einfügen	-	X
Bedienung in Echtzeit	-	(X)

Tabelle 3.2: Vergleich der einzelnen Großanzeigen-Programme

lassen. Das wird durch eine Client-/Server Architektur ermöglicht. Die Inhalte werden auf dem Server vorbereitet und können dann jederzeit an alle Client-Anzeigen gesendet werden. Hierdurch wird auch der dafür vorgesehene Einsatzzweck deutlich: Die Verteilung von Inhalten auf viele verschiedene Anzeigen. Einsatzgebiete die für dieses Programm deshalb eher in Frage kommen sind: Einkaufszentren, Schulen/Universitäten, Flughäfen und Bahnhöfe, Diskotheken, Banken oder Kinos.

Das LED Studio bietet im Vergleich zur Präsentations-Software den Vorteil, dass es wesentlich mehr Multimedia Inhalte unterstützt. Jedoch ist die Steuerbarkeit noch eingeschränkter, da das Abspielen in einer Dauerschleife erfolgt und nicht beeinflusst werden kann. Dieses

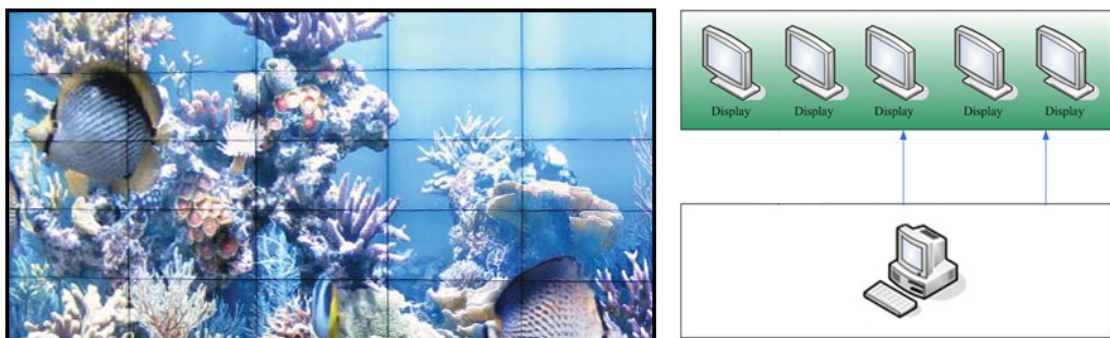


Abbildung 3.1: Über Großanzeigen-Software können einzelne Anzeigen zu einem Cluster zusammengeschlossen, oder mehrere Anzeigen getrennt voneinander bespielt werden. (Quelle: CineMassive).

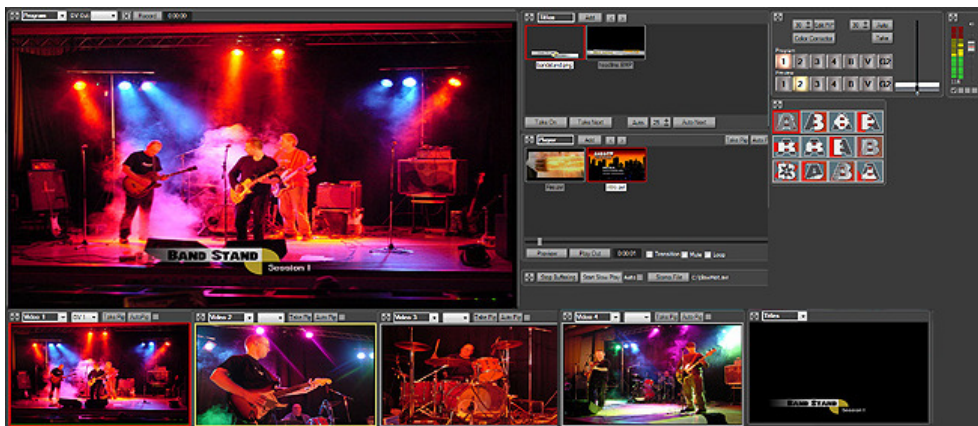


Abbildung 3.2: Playout-Systeme können Live-Video in Echtzeit mischen und bieten eine Steuerung wie bei einem Videomischer an (Quelle: avtake).

Programm ist deshalb eher für (bewegte) Diashows gedacht.

Auch hier lässt sich feststellen, dass die vorhandene Software eher für andere Bereiche und nicht für den Sport vorgesehen sind. Das LED Studio bietet zudem nur einfache Grundfunktionen und hilft dem Benutzer lediglich, schnell und einfach bewegte Inhalte auf die Videowand zu bringen.

3.2.3 Playout-Systeme

Playout-Systeme sind hauptsächlich für die Anzeige von Live-Kamerabildern vorgesehen. Eingesetzt werden diese häufig auf Konzerten oder Live-Auftritten, bei denen gleichzeitig mehrere Kameras zum Einsatz kommen. Die Kamerabilder können über die Software in Echtzeit gemischt und überblendet werden. Manche Programme bilden einen Videomischer auch eins zu eins in der Benutzeroberfläche ab. Diese Programme bieten daher die nötige Flexibilität in der Bedienung. Diese Flexibilität kommt aber häufig mit einer großen Komplexität der Bedienung daher. Es werden sehr viele Funktionen angeboten um die Videos mischen oder Bild-in-Bild Kompositionen erzeugen zu können. Diese sind überall auf der Benutzeroberfläche verteilt und bieten nur dem Experten eine Möglichkeit, die Software effektiv und effizient verwenden zu können. Diese Software ist daher auf eine ganz spezielle Zielgruppe ausgerichtet und bietet für diese auch sehr viele spezialisierte Funktionen an. Auch nach Herstellerangaben ist der Anwendungsbereich die Echtzeitmischung von Live-Videos. Betrachtet man andere Medieninhalte wie Bilder oder vorher aufgenommene Videos, stellt man fest dass diese zwar unterstützt jedoch nicht untereinander gemischt werden können. Die Anzeige ist eher als Nebenfunktion gedacht, um Ankündigungen zu machen, oder ein paar Werbeclips auszustrahlen. Videos können lediglich einzeln „ausgespielt“ werden. Bei Sportveranstaltungen wären diese Systeme grundsätzlich denkbar wenn ausschließlich Videos untereinander gemischt werden sollen. Jedoch trifft das nur beim Stadion-TV zu, bei dem Live-Kamerabilder auf Großanzeigen gebracht werden.

	Cutfour 3
Bilder	(X)
Text/Rich-Text	-
Videos	X
Externe Videoquellen	X
Videostreams	-
Flash-Inhalte	-
Anbindung an externe Datenquellen	-
Automatischer Medienwechsel	(X)
Manueller Medienwechsel	X
Animationen bei Übergang	X
Anzeigebereich anpassbar (Größe und Position)	-
Inhalte live einfügen	X
Bedienung in Echtzeit	X

Tabelle 3.3: Playout-System von avtake

Diese Software-Kategorie zeigt ihre Stärke in der Echtzeitbedienung. Externe Videoquellen lassen sich flexibel hinzufügen und jederzeit abspielen. In der Anzeigereihenfolge ist man nicht festgelegt. Zwar lassen sich auch Bilder und aufgenommene Videos in der Software hinzufügen, aber sind nicht zum Mischen wie die eigentlichen Live-Kamerabilder gedacht. Auch hier lässt sich feststellen, dass der Anwendungszweck nicht dem Benötigten entspricht. Die Art der Bedienung geht in die richtige Richtung, jedoch ist der Fokus der Software zu sehr auf den Live-Video-Bereich gelegt und die Bedienoberfläche ist für Nicht-Experten zu komplex.

Es wird jedoch keine alleinige Video- sondern eine Allround-Lösung benötigt, die viele verschiedene Medien kombinieren kann. Eine Kombination der Großleinwand- und Playout-Programme, wäre in dieser Hinsicht ein Schritt in die richtige Richtung.

3.2.4 Sportspezifische Software

Sportspezifische Software legt den Schwerpunkt der Präsentation auf den sportlichen Aspekt. Solche Sport-Lösungen sind auf das Stadion-TV zugeschnitten und bieten eine vielseitige Unterstützung für Mannschafts-Sportarten. Hier können demnach viele Informationen zum Spiel abgerufen und angezeigt werden, wie Spieler-Auswechslungen, Punktestände oder Disqualifikationen. Diese Programme sind sehr auf das Sportereignis festgelegt und bieten hier viele Funktionen an. Beispielsweise lassen sich im Voraus die einzelnen Spieler der Mannschaften festlegen und viele Informationen einspeisen, auf die später zugegriffen werden kann. Zudem wird auch immer eine Regietechnik benötigt, die das Programm steuert.

Die Präsentation von Multimedia-Inhalten ist hier dafür stark vernachlässigt. Es ist zwar möglich Playlisten für Videoclips und Einspielungen zu erstellen, die in den Pausen abgespielt werden können, aber ansonsten können im Vergleich zur Großanzeigen-Software nicht allzu viele Medientypen eingebunden werden. Dafür sind diese Programme aber auch gar



Abbildung 3.3: Sportspezifische Software erfordert zur Bildkomposition des Stadion-TV eine zusätzliche Regie (Quelle: JOY event & media).

nicht gedacht, denn sie sind auf die jeweiligen Sportarten spezialisiert und bieten in der Hinsicht eine sehr gute und große Funktionalität.

3.3 Zusammenfassung

Durch die Betrachtungen in diesem Kapitel wird klar, dass die Problemstellung durch keine der vorhandene Programme gelöst wird. Keine Software aus den gängigen Kategorien kommt den Forderungen an eine Steuerung von vielfältigen Inhalten in Echtzeit nach. Präsentationsprogramme können zwar manuell gesteuert werden, jedoch ist ein flexibles Wechseln zu bestimmten anderen Inhalten nicht problemlos möglich. Inhalte lassen sich auch nicht während einer Präsentation hinzufügen.

Die Großanzeigen-Software deckt die gewünschten multimedialen Fähigkeiten gut ab, aber ist wie auch die Präsentations-Software in der Bedienung sehr eingeschränkt.

Playout-Systeme bieten seitens der Bedienung zum Teil eine effektive Steuerung der Inhalte in Echtzeit an, jedoch ist der Fokus auf Live-Video-Quellen gelegt, die untereinander gemischt werden können. Dazu ist die Bedienung teilweise zu komplex, was die Nutzungseffizienz für Nicht-Experten sehr einschränkt.

Sportspezifische Software ist meistens sehr auf eine Sportart eingeschränkt und bietet hier sehr viele Anzeigemöglichkeiten an, wie sie auch im Fernsehen verwendet werden. Jedoch fehlt es diesen Programmen an Grundfunktionalitäten im Multimedia-Bereich.

Aus den vorangegangenen Untersuchungen wird ersichtlich, dass die vorhandene Software für einen anderen Verwendungszweck vorgesehen ist und eine andere Nutzergruppe ansprechen soll. Demnach können bisher existierende Programme die eingangs gestellten

3. VIDEOWÄNDE UND SOFTWARE

Probleme nicht lösen. Es wird dadurch deutlich, dass eine Entwicklung von neuen Konzepten und Umsetzung in einem Prototypen sinnvoll ist.

In den nächsten zwei Kapiteln werden die Grundlagen für diese Arbeit erläutert.

Kapitel 4

Live-Video Systeme

In diesem Kapitel werden Grundlagen zu den Bereichen Live-Video, Videomischer und Videowänden vermittelt, die zum Verständnis für die weitere Arbeit hilfreich sind. Es werden zunächst Möglichkeiten erörtert, welche Videosysteme in der Lage sind, Kamerabilder live auszugeben und zu verteilen. Danach werden grundlegende Begriffe zu Videomischern erklärt und die wichtigsten Funktionen erläutert. Dieses Wissen ist insbesondere wichtig, da viele Konzepte eines Videomischers auch für eine Software interessant sein könnten. Eine Zielgruppe derselben wären Medientechniker, die sich somit auch einfacher in die Software einarbeiten könnten. Im letzten Teil werden Grundlagen zu Videowänden vermittelt und erklärt, wie die Eingabesignale zur Videowand gespielt werden.

4.1 Videosysteme

Um die Kamerabilder live übertragen zu können, sind entsprechende Geräte notwendig. In diesem Teilbereich werden deshalb Möglichkeiten der Übertragung von Live-Kamerabildern gezeigt. Hierzu werden die gängigsten Möglichkeiten betrachtet. Diese Betrachtungen sind insbesondere wichtig um zu erfahren, über welchen Weg die Videodaten später in den Computer eingespeist und von einer Software abgerufen werden können.



Abbildung 4.1: Bei Videokameras oder Camcordern wird das Videosignal live ausgegeben (Quelle: Sony Corporation).



Abbildung 4.2: Durch eine interne oder externe Videokarte können Videosignale in den Computer eingespeist werden (Quelle: Blackmagic Design).

4.1.1 Videokameras und Camcorder

Die beste Möglichkeit zum Erzeugen von Live-Kamerabildern stellen Videokameras und Camcorder dar. Über diese Geräte steht das Videosignal sofort zur Verfügung und kann über verschiedene Schnittstellen ausgegeben werden. Videokameras und Camcorder unterscheiden sich lediglich darin, dass Camcorder zusätzlich einen eingebauten Videorekorder besitzen, auf dem das Videosignal parallel aufgenommen werden kann.

Wichtig für die spätere Eingabe in den Computer ist eine Betrachtung der Schnittstellen. Im professionellen Bereich werden hier sogenannte SDI-Schnittstellen (Serial Digital Interface) verwendet. Ansonsten ist an vielen Geräten eine HDMI-, FireWire- oder Cinch-Schnittstelle üblich, die oft in der Unterhaltungselektronik genutzt werden. Die Signale können beispielsweise von Videomischern direkt abgegriffen werden. Zur Nutzung mit einem Computer oder Laptop wird eine in- oder externe Videokarte benötigt (siehe Abbildung 4.2). Über diese kann dann auf die entsprechenden Videosignale von jeder beliebigen Software zugegriffen werden. Gängige Videokarten stellen viele Anschlussmöglichkeiten direkt oder über Adapter bereit, deshalb ist eine Nutzung der Videosignale am Computer problemlos möglich. Es wird jedoch eine entsprechend aktuelle Hardware benötigt, um auch HD-Videos performant verarbeiten zu können. Das ist insbesondere für ein Echtzeitsystem wichtig.

4.1.2 Netzwerk-Kameras

Netzwerk-Kameras oder auch IP-Kameras genannt, geben ihre Bilder über IP-Netzwerke weiter. Deshalb kann auf das Videosignal von jedem Computer mit eingebauter Netzwerkkarte direkt zugegriffen werden. Meistens werden Netzwerk-Kameras zur Überwachung eingesetzt, jedoch stehen aktuelle Modelle auch normalen Kameras in vielen Punkten nichts nach. Netzwerk-Kameras können sehr flexibel eingesetzt werden: Durch die Netzwerkverbindung können die Bilder in das Internet übertragen werden oder auch auf dem Veranstaltungsgelände über WLAN bequem verteilt. Nachteil ist jedoch, dass diese Kameras fest installiert und nicht für Aufnahmen an verschiedenen Orten gedacht sind. Im Sportbereich lassen sich



Abbildung 4.3: Netzwerk-Kameras werden meistens zur Überwachung eingesetzt, bieten aber eine hohe Flexibilität durch ihre Netzwerkfähigkeiten (Quelle: Axis Communications).

hier demnach keine Interviews oder ähnliche Aufnahmen machen. Hier sollten dann normale Videokameras oder Camcorder verwendet werden.

Ideal geeignet sind Netzwerk-Kameras aber, wenn ein fester Bereich, wie ein Zieleinlauf oder ein Spielfeld gefilmt werden soll. Manche Kameras bieten hier eine sogenannte PTZ-Steuerung an, die über das Netzwerk bedient werden kann. PTZ steht hier für Pan, Tilt und Zoom (schwenken, neigen, zoomen). Viele Netzwerk-Kameras zeichnen sich auch durch einen hohen Zoomfaktor aus, der auch ferngesteuert werden kann. Ein Zieleinlauf oder Spielfeld lässt sich hier demnach perfekt abdecken. Um das Videosignal von einem Computer abzurufen kann hier direkt auf den Netzwerk-Videostream der Kamera zugegriffen werden. Demnach lassen sich Netzwerk-Kameras auch gut und flexibel für Live-Kamerabilder nutzen.

Im Rahmen dieser Arbeit wird das System für eine Zusammenarbeit mit der AXIS P5534-E Kamera optimiert. Diese Kamera streamt Videos in HDTV mit einer Auflösung von 1280x720 Bildpunkten in das Netzwerk. Zusätzlich ist sie mit einem 18-fachen optischen Zoom und einer PTZ-Steuerung ausgestattet. Einsatzbereich für diese Kamera sind Rad- und Laufveranstaltungen im Zieleinlauf-Bereich.

4.2 Videomischer

Ein Videomischer wird verwendet um zwischen verschiedenen Videoquellen umzuschalten oder diese zu mischen. Über eine Bildmischung können Bild-in-Bild Kompositionen erreicht, oder Inhalte überlagert werden. Beim hin- und herschalten zwischen den verschiedenen Quellen kann zwischen dem Hartschnitt und verschiedene Übergänge (Transitions) gewählt werden. Oft unterstützen Videomischer auch Spezialeffekte mit denen das Bild manipuliert werden kann. Die Übergänge werden mit der sogenannten Fadebar erzeugt. Die Fadebar ist ein Hebel, der beim Umlegen jeweils einen Übergang erzeugt. Je nachdem wie schnell das Umschalten erfolgt wird auch der Übergang schneller oder langsamer durchgeführt. Bei manchen Videomischern kann auch eine Auto-Transition festgelegt werden, also ein automatischer Übergang nach einer festgelegten Zeit.



Abbildung 4.4: Die wichtigsten Elemente eines Videomischers (Quelle: Roland Corporation).

Als wichtigste Steuerelemente eines Videomischers sind zu nennen (siehe Abbildung 4.4):

- Bus - Der Bus ist eine Anordnung von Knöpfen, die für die verschiedenen Eingangssignale stehen. Wird ein Knopf gedrückt, so wird das Videosignal nach einem Übergang auf den Ausgang des Videomischers geschaltet.
- Fadebar - Mit der Fadebar oder auch T-Hebel genannt, kann ein Übergang zwischen dem aktuellen und als nächstes gewähltem Bus erzeugt werden. Die Art des Übergangs kann bei den Übergangseffekten eingestellt werden. Gängige Übergänge sind Überblendungen und Wisch-Effekte.
- Vorschaufenster - Zur Vorschau bieten manche All-In-One-Geräte einen kleinen Monitor an, auf dem das aktuell ausgegebene Bild angezeigt wird (siehe Abbildung 1.2). Ansonsten kann die Bildvorschau immer über einen externen Monitor angesehen werden.
- Spezialeffekte - Viele Videomischer bieten diverse Spezialeffekte an, die in Echtzeit auf das Ausgabevideo angewendet werden können. Beispiele hierfür sind Farbeffekte, Helligkeitseinstellungen oder Bildspiegelungen.

4.3 Videowände

Beim technischen Aufbau von Videowänden ist zwischen LCD-, Plasma- und LED-Technik zu unterscheiden. Heutzutage werden im Sportbereich hauptsächlich LED-Videowände eingesetzt [PHPP11a]. Die LED Technik wird für größere Anzeigen auch oft bevorzugt, weil LCD und Plasma im Vergleich wesentlich teurer sind. LED-Wände stehen der LCD- oder Plasma-Technik trotz dessen in nichts nach. Zu sehen ist das auch in der Unterhaltungselektronik, im Bereich der Fernseher — die Zahl der LED-Geräte nimmt stetig zu. Welche Technik auf den Veranstaltungen jeweils zum Einsatz kommt, kann letztendlich verschiedene



Abbildung 4.5: Videowände werden aus einzelnen Modulen zusammengesetzt. (Quelle: Ex-promo A/S, Medien Werkzeuge).

Entscheidungskriterien haben. Vom grundsätzlichen Aufbau und der Ansteuerung verhalten sich die verschiedenen Techniken aber ähnlich.

Videowände bestehen je nach Größe nicht aus einem zusammenhängenden Teil, sondern aus mehreren Modulen, die zusammengeschaltet werden (siehe Abbildung 4.5). Die Ansteuerung erfolgt deshalb immer über einen Videowand Controller, der die einzelnen Module verwaltet und für die Bildausgabe verantwortlich ist. Dieser Controller kann in einem zusätzlichen Videocontroller als externe Lösung genutzt werden oder auch direkt als Karte im Computer.

Wer schon mal einen Beamer oder einen zweiten Monitor betrieben hat, weiß somit auch, wie man die Anzeige auf einer Videowand einrichtet. Hierzu muss lediglich die Monitoranzeige gespiegelt (geklont), oder auf die Videowand als zweiten Bildschirm erweitert werden. Je nach Einstellung des Videowand Controllers greift die Videowand das gesamte Bild oder nur einen kleineren Bereich des Bildschirms ab.

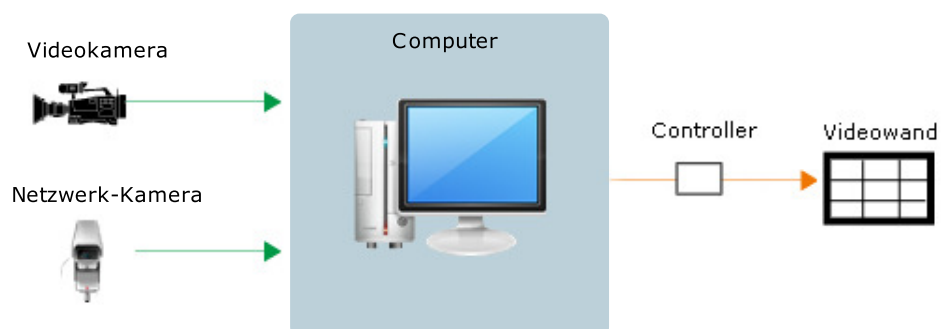


Abbildung 4.6: Bei der Verwendung des Computers als Eingabequelle für die Videowand, kann ein interner oder externer Controller für die Videowand genutzt werden.

4.4 Zusammenfassung

In diesem Kapitel wurde erläutert, dass sowohl Videokameras und Camcorder, als auch Netzwerk-Kameras eine gute Lösung für Sportveranstaltungen darstellen. Alle diese Kameratypen verfügen über eine Live-Ausgabe der Bilder. Je nach Anwendungszweck kann entschieden werden, welche Lösung jeweils die Beste ist. Zu beachten ist jedoch, dass für Videokameras und Camcorder eine zusätzliche Erweiterungskarte für den Computer (in-/ oder extern) benötigt wird. Netzwerk-Kameras benötigen lediglich eine Netzwerkkarte, die heute fast jeder Computer besitzt. Jedoch sind Netzwerk-Kameras eher für eine Installation an einem festen Ort vorgesehen, beispielsweise einem Zieleinlauf bei Laufveranstaltungen. Durch eine PTZ-Steuerung können einige Modelle aber ferngesteuert werden.

Auf Seiten der Videomischer wurden Grundbegriffe wie der Bus oder die Fadebar näher erläutert. Diese werden in späteren Kapiteln noch eine Rolle spielen.

Letztendlich wurde geklärt was beim Einsatz von Videowänden zu beachten ist. Diese bestehen aus einzelnen Modulen und benötigen demnach immer einen Videowand Controller (in-/ oder extern), der die Ansteuerung ermöglicht. Wichtig ist auch, dass sich die Videowand wie ein zusätzlicher Monitor verhält und entweder den gesamten Bildschirm oder nur einen Teilbereich abgreift.

Alle in diesem Kapitel erläuterten Dinge sind Grundlagen für die später ausgeführte Methodik. Im nächsten Kapitel werden Grundlagen im Bereich der Programmierung näher betrachtet.

Kapitel 5

Grundlagen der Programmierung

In diesem Kapitel wird auf Grundlagen der Programmierung eingegangen, auf die in der Entwicklungsphase zurückgegriffen wird. Hierzu werden Flash und das Flex Entwicklungsframework näher betrachtet. Wichtig ist hier insbesondere die Leistung der Programmiersprache in Hinblick auf Echtzeit und Multimedia-Anwendungen und ob sie in der Lage ist, die gewünschten Anforderungen umzusetzen. Im darauf folgenden Teil werden kurz Actionscript 3 und MXML betrachtet, die zur Programmierung verwendet werden sollen. Zur einfacheren Umsetzung von Animationen und Übergängen, die später eingesetzt werden sollen wird die „Greensock Tweening Platform“ im letzten Teil dieses Kapitels beschrieben.

5.1 Einleitung

Voraussetzung für ein System das in Echtzeit gesteuert werden soll ist auch eine Laufzeitumgebung, die Befehle in Echtzeit umsetzen kann. Somit werden auch an die Programmierung gezielte Anforderungen gestellt. Diese lassen sich wie folgt zusammenfassen:

- Performance — Das System muss sich in Echtzeit bedienen lassen und somit die Befehle auch in unmittelbarer Zeit umsetzen können. Auch beim Ablauf der Präsentation muss ein flüssiges Bild zustande kommen.
- Grafische Animationen — Um einen professionellen Effekt zu erzielen soll beim Medienwechsel eine Übergangsanimation stattfinden. Diese muss sich in der Programmiersprache möglichst einfach umsetzen lassen.
- Video — Die Programmiersprache muss in der Lage sein, Videos von einem Datenträger abzuspielen. Zusätzlich soll die Möglichkeit vorhanden sein, Videostreams aus dem Netzwerk und Videosignale von einer Videokarte entgegen zu nehmen. Diese Videos sollen bis zu einer Auflösung in Full-HD ruckelfrei und problemlos abgespielt werden können.
- Steuerelemente — Für den Prototyp sollen gegebenenfalls eigene Steuerelemente entwickelt werden. Die Programmiersprache sollte dazu ebenfalls in der Lage sein.



Abbildung 5.1: Durch die Flash- und Flex-Technologie können AIR-Anwendungen erstellt werden, die auf dem Desktop laufen (Quelle: Adobe).

5.2 Flash/Flex

Adobe Flash¹ ist eine Entwicklungsumgebung, die zur Erstellung multimedialer, interaktiver Inhalte verwendet wird. Ursprünglich wurden hiermit nur Inhalte für den Adobe Flash Player entwickelt, die im Web-Browser abgespielt werden können. Mit der Einführung von Flex, einem zusätzlichen Entwicklungsframework von Adobe, wurden die Fähigkeiten von Flash im Bereich der Rich Internet Applications (RIAs) hier wesentlich verbessert. Rich Internet Applications sind Programme, die sowohl im Internet, als auch auf dem Desktop ausgeführt werden können. Diese Programme werden dann als AIR-Anwendungen bezeichnet und nutzen die AIR-Laufzeitumgebung (Adobe Integrated Runtime) von Adobe. Diese Laufzeitumgebung ähnelt der von Java Anwendungen und nutzt die Programmiersprache Actionscript 3. Wie auch Java sind AIR-Anwendungen plattformunabhängig und können bisher unter Windows, Macintosh, Linux und Android ausgeführt werden.

Performance

AIR-Anwendungen laufen wie herkömmliche Programme auf dem Desktop, deshalb werden ihnen mehr Ressourcen vom Betriebssystem zugesprochen als Flash Anwendungen im Browser. Zum Testen der Performance im Bezug auf Grafik und Animationen wird ein Benchmark hinzugezogen. Der „Bubblemark animation test“² ist eigentlich zum Vergleich von RIA-Frameworks in verschiedenen Web-Browsern gedacht. Jedoch lassen sich Java- und Adobe Flex-Anwendungen auch als Desktopanwendungen kompilieren. Deshalb sollen diese beiden miteinander verglichen werden. Der Quellcode kann direkt von der Website heruntergeladen und in die jeweiligen Entwicklungsumgebungen importiert und auf Desktopanwendungen umgesetzt werden.

Im direkten Vergleich von Java (Swing) und AIR ergaben sich folgende Ergebnisse (siehe Abbildung: 5.2): Java schneidet im Vergleich zu AIR bei einer Animation von 100 Bällen, deutlich schlechter ab. Hier erreicht es nur eine Framerate von 64 Bildern pro Sekunde (fps), hingegen erreicht AIR sogar 176fps. Auch bei 1000 Bällen ist noch ein geringer Performance-Unterschied vorhanden (+4fps). Die Leistung von AIR-Anwendungen sollte hier demnach ausreichen, da sie im Vergleich zu Java-Anwendungen im grafischen Bereich deutlich besser abschneidet.

¹<http://www.adobe.com/de>

²<http://bubblemark.com>

Bubblemark Animation Test

Hardware: Intel Core2Duo E8400 2x3.00GHz; 2GB RAM; ATI Radeon HD 3870
Software/Treiber: Windows XP SP3; ATI Catalyst 10.10
Flash Version: Flash Player 10.3.181.22; AIR 2.6
Java Version: 6 Update 24

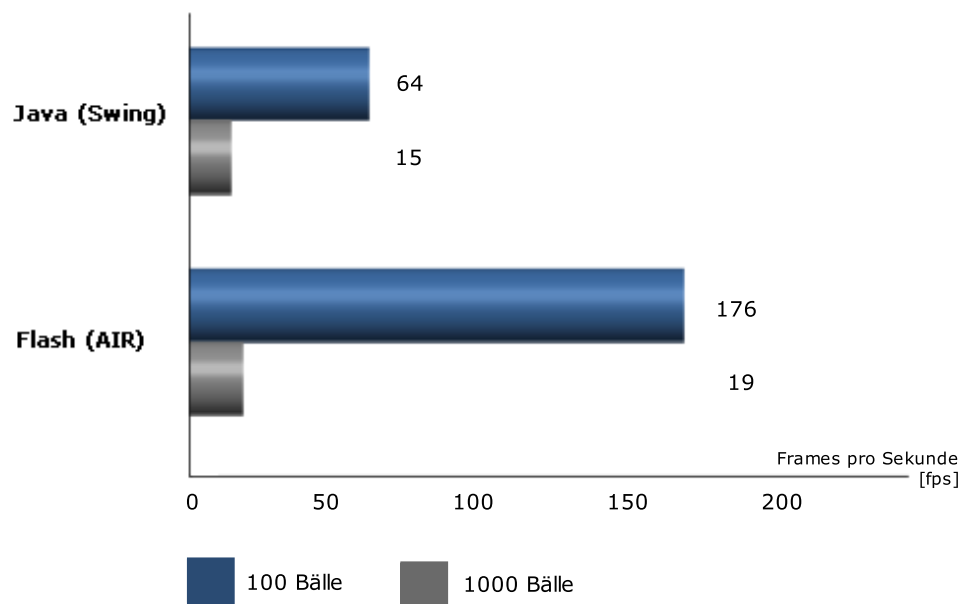


Abbildung 5.2: Java schneidet im Vergleich zu AIR deutlich schlechter ab.

Grafische Animationen

Flash wurde ursprünglich als reines Animationsprogramm verwendet. Deshalb sind die Funktionen im grafischen Bereich inzwischen sehr ausgereift und einfach zu verwenden. Es werden im Vergleich zu C oder C++ keine zusätzlichen Bibliotheken benötigt. Aufgrund seiner multimedialen Möglichkeiten ist Flash also bestens für ein Präsentationsprogramm geeignet.

Video

Videodateien werden von Flash in einigen Formaten unterstützt (bevorzugt Flash Video und Quicktime). Im Internet kommt bisher fast ausschließlich der Flash Player zur Anzeige von Videoinhalten auf Videoportalen zum Einsatz. In den neusten Versionen der AIR-Technologie wurde die H.264-Hardwarebeschleunigung für Windows eingeführt, die auch die Unterstützung von Videos in HD-Auflösungen beinhaltet.

Steuerelemente

Im Adobe Flex Entwicklungsframework lassen sich die Komponenten sehr einfach verändern und auf die eigenen Bedürfnisse anpassen. Über sogenannte Skins wird eine grundlegende Änderung des visuellen Erscheinungsbildes von Flex-Komponenten ermöglicht. Auch aus dieser Hinsicht ist Flash demnach gut geeignet.

5.3 GreenSock Tweening Platform

Über die kostenlose Tweening Platform von GreenSock werden Animationen in Kombination mit Flash und Flex noch einfacher gemacht. Die Tweening Platform besteht aus zwei Komponenten, dem Tween und der Timeline, die in jeweils verschiedenen Versionen verfügbar sind. Tween zeichnet sich durch seine Flexibilität und hohe Performance aus, die von GreenSock sogar höher als die von Adobes Standard Tween-Animation angegeben wird³. Durch Timeline können verschiedene Tween-Animationen nacheinander angeordnet und abgespielt werden. Dadurch werden viele Animationsfunktionen von Adobe stark vereinfacht. Wichtige Funktionen, die im Bezug auf ein Präsentationsprogramm benötigt und auch von der Tweening Platform unterstützt werden sind:

- Bewegungstweens
- Einblend-/Ausblendeffekte
- Flexible Kontrolle über die Animationen

Die Tweening Platform wird von vielen großen Unternehmen bevorzugt. Hier lassen sich beispielsweise nennen: Mercedes-Benz, Saab, Lexus, Motorola oder auch Lego.

5.4 Actionscript und MXML

Actionscript 3.0 (kurz AS3) ist eine objektorientierte Programmiersprache, die es ermöglicht, interaktive Anwendungen in Flash, Flex und AIR zu erzeugen. Sie ähnelt in vielen Teilen stark Java⁴. Einen genaueren Überblick über die AS3 Sprach- und Syntax-Elemente kann auf der Adobe Website eingesehen werden⁵. Dort findet sich auch die Sprachreferenz⁶ zu AS3 wieder.

MXML (Multimedia eXtensibleMarkup Language) ist eine XML-Auszeichnungssprache für Flex-Inhalte. Mit MXML können ähnlich wie in HTML, visuelle Elemente beschrieben werden. Hier vereinfacht MXML die Programmierung in Actionscript, da die einzelnen visuellen Elemente nicht programmiert werden müssen, sondern durch verschiedene Tags in einer

³<http://www.greensock.com/tweening-speed-test>

⁴<http://www.java.com>

⁵http://help.adobe.com/de_DE/ActionScript/3.0_ProgrammingAS3

⁶http://help.adobe.com/de_DE/AS3LCR/Flash_10.0/index.html

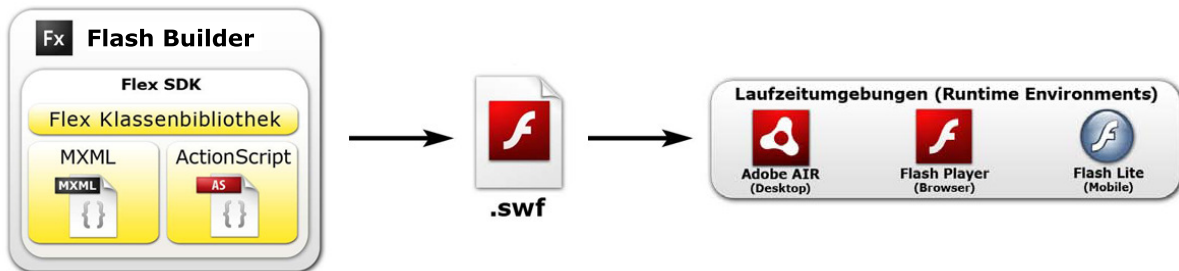


Abbildung 5.3: Über den Flash Builder können durch MXML und Actionscript interaktive Anwendungen erzeugt werden. Diese werden dann in eine Flash Datei kompiliert und können später in verschiedenen Laufzeitumgebungen wie AIR verwendet werden (Quelle: Adobe).

MXML-Datei festgelegt werden können. Jede MXML-Komponente repräsentiert eine AS3-Klasse.

In Adobes Entwicklungsumgebung, dem Flash Builder, können MXML und Actionscript Elemente kombiniert werden. Gängig ist es, Bedienelemente in MXML zu realisieren und die Programmlogik in Actionscript umzusetzen. Beispiele für MXML und Actionscript sind im Anhang B zu finden.

5.5 Zusammenfassung

In diesem Kapitel wurden die Grundlagen zur Programmierung geklärt. Adobe bietet mit der AIR-Technologie die Möglichkeit an, interaktive Anwendungen in Flash zu entwickeln, die dann auf dem Desktop ausgeführt werden können. Flash überzeugt einerseits in den grafischen Fähigkeiten, als auch in Sachen Performance. Videos werden auch in großen HD-Auflösungen unterstützt. Zudem läuft die AIR-Technologie auf den gängigsten Betriebssystemen. Parallel dazu wurde die zu verwendende Programmiersprache Actionscript 3 und die Auszeichnungssprache MXML erwähnt. Diese beiden können im Flash Builder von Adobe zu anspruchsvollen, interaktiven Anwendungen kombiniert werden. Über die GreenSock Tweening Plattform sollen die späteren Animationen und Übergänge realisiert werden.

Die vermittelten Grundlagen werden bei der Ausführung der Methodik in den nächsten Kapiteln unterstützend sein.

Kapitel 6

Konzeptentwicklung

Das nachfolgende Kapitel beschäftigt sich mit der Entwicklung benutzerseitiger Konzepte für eine Software, die multimediale Inhalte auf einer Videowand darstellen soll. Insbesondere beachtet wird der spätere Einsatzort der Software, hier Sportveranstaltungen. Der erste Schritt ist eine Analyse der Anforderungen, die sich an eine solche Software als Mensch-Computer-Schnittstelle stellen. Diese werden von verschiedenen Seiten aus betrachtet: Echtzeit, Sport und Multimedia.

Im nächsten Schritt werden aus den Anforderungen einzelne Konzepte für die spätere Benutzung entwickelt. Hierbei soll insbesondere das Ziel der Gebrauchstauglichkeit beachtet werden. Bevor die Benutzerkonzepte entwickelt werden wird eine Analyse der späteren Nutzergruppen durchgeführt. Im letzten Teil wird der Funktionsumfang der Software über die Use Cases (Anwendungsfälle) festgelegt.

6.1 Einleitung

Betrachtet man den Stand der Forschung so stellt man fest, dass viele Untersuchungen im Bereich Usability existieren die gezielte Vorgaben an eine gebrauchstaugliche Software stellen. Vorgaben für Usability sind, dass der Benutzer die Software leicht erlernen und seine Ziele effektiv, effizient und zufriedenstellend erreichen kann. Diese Vorgaben müssen auch für die im Rahmen dieser Arbeit entwickelten Konzepte beachtet werden und stellen die Grundlage für diese dar. Deshalb wird auf diese Konzepte aufgebaut. Um benutzerseitige Konzepte für eine Software zu entwickeln gibt es verschiedene Möglichkeiten. Aufgrund der gestellten Zielvorgaben ist folgendes Vorgehen sinnvoll: Man orientiert sich an funktionierenden Konzepten aus der realen Welt. In der Programmierung findet häufig eine Abstraktion von realen Objekten statt — das heißt, man versucht Merkmale und Fähigkeiten von existierenden Objekten in die Software zu übertragen. Diese Abstraktion soll, wenn möglich, in diesem Kapitel auch durchgeführt werden. Bevor diese Konzepte jedoch entwickelt werden können ist eine Analyse der späteren Nutzergruppen notwendig. Bestehen diese nur aus Fachpersonal, kann eine Software anders umgesetzt werden als wenn viele unterschiedliche Nutzergruppen das System nutzen sollen.

6.2 Analyse der Anforderungen

Auf Sportveranstaltungen ist eine der wichtigsten Anforderungen an eine Software, die multimediale Inhalte auf einer Videowand präsentieren soll, eine Bedienung in Echtzeit. Einfacher gesprochen: Die Software soll es ermöglichen, die angezeigten Inhalte jederzeit auf Knopfdruck wechseln zu lassen und das live. Eine solche Software ist vergleichbar mit einem Videomischer: Jede Aktion ist direkt für den Betrachter auf dem Anzeigemedium sichtbar. Es wird ersichtlich, dass die Mensch-Computer-Interaktion daher eine große Rolle spielt und besonders beachtet werden muss.

Aus der Anforderung nach Echtzeit resultieren demnach folgende Schlussfolgerungen:

1. Jede Bedienaktion ist kritisch in ihrer Ausführung und wird für den Betrachter direkt sichtbar. Bedienfehler können demnach einen großen Schaden anrichten und wirken für die Betrachter unprofessionell. Durch den Sport als Anwendungsgebiet ist Stress auch ein wichtiger Faktor, der berücksichtigt werden muss. Bedienfehler sollten daher durch gute Konzepte und eine effektive Benutzeroberfläche vermieden werden.
2. Benutzerseitige Fehler können schneller auftreten — Ähnlich wie Bedienfehler sind Fehler, die der Anwender neben der Bedienung der Software noch machen kann. Startet er etwa im Hintergrund noch andere Programme, könnte es vorkommen das diese — wenn auch unbeabsichtigt — auf der Videowand auftauchen. Wie in den Grundlagen schon erwähnt, verhält sich eine Videowand wie ein zweiter Bildschirm. Demnach könnte der Anwender aus Versehen die Videowandfläche überdecken und Fehlbilder erzeugen. Somit ist es wichtig, dass nur Inhalte auf der Videowand auftauchen, die auch dafür vorgesehen sind. Diese Fehlerquelle ist potenziell gegeben, da die Präsentation in Echtzeit ablaufen soll.
3. Der Benutzer muss die volle Kontrolle über die Präsentation haben — Vor allen bei Sportveranstaltungen muss schnell agiert werden können. Es muss also direkt auf gewünschte Medienobjekte oder Live-Kamerabilder umgeschaltet werden können, egal wo man sich gerade in der Präsentation befindet. Dieses Wechseln soll einerseits durch einen einfachen Knopfdruck und unsichtbar für den Betrachter erfolgen können. Andererseits aber auch manuell beeinflussbar sein — in Geschwindigkeit als auch vom Startzeitpunkt.
4. Die Inhalte der Präsentation müssen kontrollierbar sein — Meistens ist die Liste der anzuzeigenden Medien nicht dauerhaft und muss während einer Präsentation verändert werden. Es sollen sich also auch im laufenden Betrieb Medien beliebig hinzufügen oder auch aus der Anzeigeliste entfernen lassen. Die Reihenfolge der einzelnen Medien soll auch flexibel beeinflussbar sein. Diese sollen sich je nach Bedürfnis umtauschen lassen.
5. Die Verzögerung zwischen Bedienaktion und Umsetzung der Aktion auf der Anzeige muss gering sein — Soll ein interaktives System in Echtzeit gesteuert werden, so darf die Bildausgabe auf dem Anzeigemedium (hier der Videowand) mit minimalster Verzögerung auf Bedienaktionen angezeigt werden. Kritisch könnte diese Verzögerung

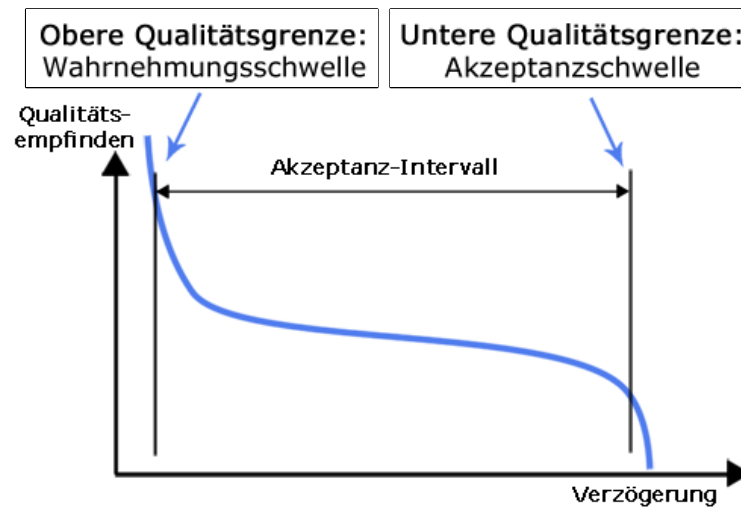


Abbildung 6.1: Die Verzögerung der Bildausgabe in Folge der Bedienaktion muss innerhalb eines bestimmten Intervalls liegen.

bei Medienobjekten mit hohen Anforderungen an die Hardware sein, die also viele Ressourcen ausnutzen, beispielsweise Full-HD-Videos und Videostreams. Die Software muss hier also innerhalb eines gewissen Zeitintervalls reagieren, das vom Nutzer als akzeptabel angesehen wird (siehe Abbildung 6.1). Nach wissenschaftlichen Untersuchungen liegt diese Schwelle durchschnittlich bei 78 ± 14 ms [Zub02].

6. Auch Programmfehler wirken sich kritisch auf das Gesamtbild der Präsentation und auf das Verhältnis zwischen Software und Benutzer aus — Falsches Verhalten oder Programmfehler (Bugs) sollten deshalb verhindert werden. Insbesondere bei Echtzeitanwendungen ist das ein kritischer und demnach wichtiger Punkt. Bei Fehlverhalten der Software sinkt das Vertrauen der Benutzer und die Motivation, die Software auch weiterhin benutzen zu wollen (vgl. [SP05]).

Neben den Anforderungen in Bezug auf Echtzeit stellen sich auch noch weitere Anforderungen unter besonderem sportlichen Aspekt, die wie folgt betrachtet werden:

1. Eine Videowand kann eine Anzeigetafel für Spielstände und Ergebnisse einfach ersetzen — Es sollen deshalb im Prototypen Live-Ergebnisse auf Basis von Rad- und Laufveranstaltungen entwickelt werden, also eine Top Ten Anzeigeliste mit Platzierung, Name und Zeit des Athleten. Für die Ergebnisabfrage muss Kontakt zu einem externen Datenanbieter aufgenommen werden, über den dann die Daten abgerufen werden können.
2. Sportveranstaltungen leben von der Unterstützung der Sponsoren. Diese werden deshalb oft auf der Videowand präsentiert — Es ist deshalb sinnvoll, ein beliebiges Logo

dauerhaft anzeigen zu lassen. Somit soll sich ein Logo, beispielsweise des Hauptsponsors, an einer festen Position im Bild einbinden lassen (vergleichbar wie die Senderlogos im Fernsehen).

Betrachtet man den multimedialen Aspekt, ergeben sich noch folgende Schlussfolgerungen:

1. Programme für Videowände unterstützen eine große Vielfalt an Medienformaten — Um diesen Programmen in nichts nachzustehen, soll ebenfalls ein möglichst großes Spektrum an Mediendateien angezeigt werden können. Unterstützt werden müssen demnach mindestens: Bilder, Text/Rich-Text, Videos und Flash-Inhalte.
2. Das Sportgeschehen auf der Veranstaltung ist der Kernpunkt um den es sich dreht und soll deshalb im Vordergrund stehen — Live-Videos sollen deshalb über Videostreams oder externe Videoquellen eingebunden und angezeigt werden können. Über die Möglichkeiten wurde bereits in Kapitel 4.1 gesprochen.

Die Grundsätze der Dialoggestaltung, nach DIN EN ISO 9241-110 [92406], wurden bereits in vorhergehenden Kapiteln erwähnt. Diese zeichnen benutzerfreundliche Systeme aus und unterstützen demnach das angestrebte Ziel nach Effizienz, Effektivität und Anwenderzufriedenheit (genauer siehe Tabelle 2.1). Deshalb sollen diese in den nachfolgenden Untersuchungen auch beachtet werden:

- Aufgabenangemessenheit
- Selbstbeschreibungsfähigkeit
- Steuerbarkeit
- Erwartungskonformität
- Fehlertoleranz
- Individualisierbarkeit
- Lernförderlichkeit

6.3 Benutzerseitige Konzeptentwicklung

In diesem Teilkapitel stellt sich die Frage, wie der Mensch am Besten mit einer Software zurecht kommt, wenn Echtzeit eine Rolle spielt. Um diese Frage beantworten zu können, müssen im ersten Schritt die späteren Nutzergruppen der Software näher identifiziert werden. Für die Benutzung eines solchen Systems kommen grundsätzlich zwei Nutzergruppen in Frage:

1. Medientechniker — Wird eine Videowand gemietet, so kann oft auch eine zusätzliche Regietechnik gebucht werden. Der typische Benutzer ist hier demnach ein Medientechniker oder Fachpersonal mit einer ähnlichen Ausbildung. Diese Personengruppe verfügt über Fachwissen in den Bereichen Videomischer und Videotechnik. Er kennt sich also mit Bedienkonzepten bestens aus.

2. Veranstalter — Bucht der Veranstalter eine Videowand ohne Regie, so ist er auf sich allein gestellt. Er bekommt die wichtigsten Punkte zur Bedienung gezeigt und wie genau er welche Geräte an die Videowand anschließen kann. In technischen Dingen sind meistens kein großen Vorkenntnisse vorhanden, der Benutzer ist hier demnach Laie.

Da die Anforderungen und Nutzergruppen jetzt festgelegt sind, können im nächsten Schritt Konzepte für eine passende Benutzerschnittstelle entwickelt werden. Da die Nutzergruppen sich sehr stark in ihrem Vorwissen unterscheiden, muss eine entsprechendes Konzept entwickelt werden, das für beide Gruppen angepasst ist. Betrachtet man die Aufgaben und den Anwendungskontext der Benutzer, so spielt die Echtzeitbedienung eine entscheidende Rolle. Hier ist es deswegen sinnvoll, sich an funktionierenden Konzepten und Systemen der realen Medienwelt zu orientieren. Bestenfalls an solchen, die schon eine Bedienung in Echtzeit nutzen.

Ein zutreffendes System ist der schon oft erwähnte Videomischer. Über ihn können Überblendungen von verschiedenen Quellen in Echtzeit gemacht werden. Ein Präsentationsprogramm ähnelt auch in vieler Hinsicht einem solchen Gerät. In der Medienwelt hat sich dieser durchgesetzt und wurde im Laufe der Jahre für eine effektive und effiziente Bedienung optimiert. Für die erste Nutzergruppe, die Medientechniker, wäre eine digitale Umsetzung eines Videomischers sehr intuitiv. Sie könnten vorhandenes Wissen auf die Software übertragen und die meisten Funktionen direkt nutzen. Für den Veranstalter als Laien wären sicherlich viele Funktionen unverständlich und der Aufbau auf den ersten Blick zu unübersichtlich. Hier kann aber ein Kompromiss gefunden werden — denn softwareseitig werden gar nicht alle Funktionen des Videomischers benötigt und es kann Vieles vereinfacht werden. Eine ledigliche Orientierung an den grundsätzlichen Funktionen und der Arbeitsweise eines Videomischers ist deshalb eher sinnvoll. Wie in Kapitel 3.2 bereits erwähnt, wird der Videomischer in verschiedenen Playout-System ebenfalls direkt umgesetzt. Die Funktionalität der Bedienung geht hier demnach schon in die richtige Richtung. Aber wie auch Videomischer, sind diese Programme ausschließlich auf die Mischung von (Live-)Videoquellen spezialisiert. Andere Inhalte wie Bilder oder vorher aufgenommene Videos können nur begrenzt genutzt werden.

Die alleinige Umsetzung eines Videomischers in eine Software ist demnach für den gewünschten Anwendungszweck nicht ausreichend. Überträgt man jedoch das grundsätzliche Bedienkonzept eines Videomischers in eine Software und erweitert dieses durch die zusätzlichen multimedialen Fähigkeiten eines Computers, so kann eine gewünschte Funktionalität erreicht werden. Als Eingabequelle ist man nicht auf externe Videos beschränkt, sondern kann alle Medienformate wählen, die der Computer (oder die jeweilige Programmierungsumgebung) unterstützt.

Bezogen auf die Arbeitsweise eines Videomischers könnten folgende Bedienelemente umgesetzt werden:

- Bus — Die einzelnen Knöpfe stehen in diesem Fall nicht für die Videoquellen, sondern



Abbildung 6.2: Prototyp der Benutzeroberfläche einer stark vereinfachten Videomischer-Software.

für die Eingabemedien. Ein Knopf kann also jederzeit mit einem beliebigen Medium (Bild, Video, Text, Flash, externe Videoquelle) belegt werden.

- Faderbar — Die Faderbar stellt hier auch wieder ein zentrales Steuerelement dar, mit dem Übergänge zwischen den Medien erzeugt werden können.
- Vorschaufenster — Ist sonst nur in großen Videomischern enthalten: Das Vorschaufenster. In einer Software unterstützt es den Benutzer zusätzlich und zeigt an, was momentan auf der Videowand angezeigt wird.

Parallel zur Faderbar sollten auch verschiedene Übergangseffekte zur Auswahl stehen, um einen guten Präsentationseffekt erzielen zu können. Auch eine Umsetzung der automatischen Übergänge (Auto-Transition) ist durchaus sinnvoll, damit das Programm auch eigenständig laufen kann.

Besonders bei einer Software werden viele Eingabemedien üblich sein, beispielsweise wenn der Benutzer viele Bilder und Videos präsentieren möchte. Hier sind Markierungen am Bus hilfreich, die anzeigen, welches Medium gerade aktiv ist und welches als nächstes angezeigt werden soll. Diese Markierungen sollen die Bedienung zusätzlich unterstützen.

Überträgt man das Konzept eines Videomischers auf die durch die Zielvorgaben festgelegte Software, so werden viele andere Funktionen nicht benötigt und sind teilweise in der Software-Welt auch gar nicht notwendig. Auch auf die Einführung von zusätzlichen Funktionen sollte generell verzichtet werden, wenn diese nicht unbedingt notwendig sind. Wenn

die Software von allen Nutzergruppen als effektiv und effizient angesehen werden soll, ist weniger oft mehr.

Beispiel (angelehnt an [RF10]): *Stellen Sie sich eine Kaffeemaschine vor, die gleichzeitig Eier kochen kann. Welche Zielgruppe erreichen Sie mit diesem Produkt:*

1.) *Kaffee-Liebhaber*

2.) *Eier-Liebhaber*

3.) *Beide Zielgruppen: Sowohl Kaffee- als auch Eier-Liebhaber*

4.) *Die Schnittmenge: Jene Leute, die bevorzugt Kaffee zusammen mit gekochten Eiern zum Frühstück genießen.*

Zusatzfrage: Wie wird die Dauer der Kaffee-Brühzeit auf die Dauer der Eier-Kochzeit abgestimmt?

Das Beispiel soll den Aspekt verdeutlichen: Produkte werden häufig mit vielen Funktionen ausgestattet, um möglichst viele Nutzergruppen anzusprechen. Mehr Funktionen müssen aber in der Regel durch eine höhere Komplexität in der Bedienung erkauft werden.

Grundsätzlich sollte demnach eine einfache Benutzeroberfläche geschaffen werden, die auf wesentliche Funktionen beschränkt ist. Diese werden im nächsten Teilkapitel durch die Use Cases noch genauer festgelegt. Funktionen eines Videomischers, die beispielsweise nicht benötigt werden, sind sämtliche Spezialeffekte, die auf das Video angewendet werden können. Im Studiobereich machen diese Effekte durchaus Sinn, jedoch werden sie auf Sportveranstaltungen keine Verwendung finden. Auch auf eine Picture-in-Picture-Funktion soll erst verzichtet werden.

Neben dem Videomischer als Grundkonzept sollten für eine Echtzeitsoftware noch andere Punkte beachtet werden. Das Ziel ist es, die Bedienung eines solchen Systems für alle Nutzergruppen möglichst einfach zu gestalten. Für eine schnelle und effiziente Steuerung ist wichtig, dass Menüpunkte und Funktionen mit wenigen Klicks erreicht werden können. Es macht bei einer so zeitkritischen Software keinen Sinn, sich durch komplexe Menüstrukturen klicken zu müssen. Generell soll auch auf eine komplexe Benutzeroberfläche verzichtet und nur Steuerelemente auf der Haupt-Benutzeroberfläche für die wichtigsten Bedienaktionen angeboten werden. Erweiterte Einstellungen können in Kontextmenüs ausgelagert werden, da sie nicht so häufig verändert werden.

Gerade bei der Bedienung in Echtzeit treten zu bestimmten Steuerelementen oft Fragen auf. Es wird vergessen wofür bestimmte Knöpfe genutzt werden können. Nachlesen in der Hilfe dauert bei einer solchen Anwendung jedoch zu lange, denn Funktionen werden meistens direkt benötigt. Deshalb sollen Hinweise zur Benutzung des jeweiligen Steuerelements direkt in der Software angezeigt werden. Hierdurch soll der Benutzer so gut wie möglich unterstützt werden.

Das Grundkonzept des Videomischers als Benutzeroberfläche bietet eine gute Grundlage und soll deshalb prototypisch in Softwarekonzepten umgesetzt werden. Die Rahmenbedingungen dazu bilden die Use Cases, die den Funktionsumfang festlegen. Diese werden im nächsten Schritt genauer beschrieben.

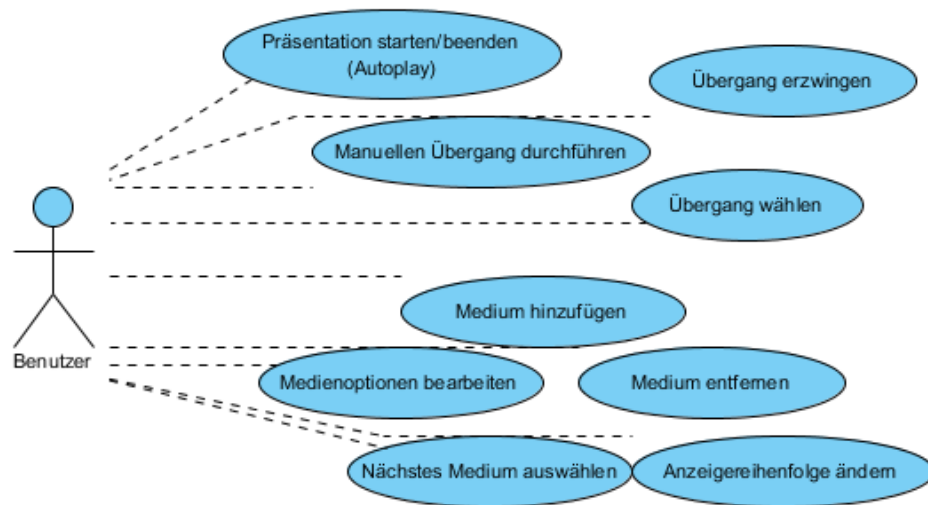


Abbildung 6.3: Hauptfunktionen einer Videomischer-Software

6.3.1 Use Cases

Über Use Cases werden die geplanten Funktionalitäten eines Systems näher beschrieben. Wie bereits erläutert, soll die Benutzeroberfläche der Software einem Videomischer nahe kommen. Jedoch können auch viele Funktionen weggelassen werden, weil sie nicht benötigt werden. Andere Funktionen hingegen können in Menüs verlagert werden, weil sie nicht ständig verwendet werden. Die Funktionen werden deshalb im Rahmen dieser Arbeit in zwei Bereiche aufgeteilt:

1. Hauptfunktionen — Diese Funktionen werden bei der Bedienung häufig benötigt und müssen deshalb direkt über die Benutzeroberfläche zugänglich sein.
2. Nebenfunktionen — Diese Funktionen sind ebenfalls wichtig, werden aber nicht so häufig verwendet. Beispielsweise muss die Größe der Anzeigefläche nur einmal bei Programmstart angepasst werden und verändert sich dann nicht mehr.

In Abbildung 6.3 sind die Hauptfunktionen zu sehen. Diese gliedern sich in zwei große Bereiche: Der erste Bereich bezieht sich auf die Präsentation der Mediendateien. Die Präsentation soll eigenständig laufen können, sprich einen automatischen Modus haben, in dem die Medien in einer festgelegten Reihenfolge angezeigt werden. Das Gegenstück dazu wäre der manuelle Übergang, der über die Fadebar erzeugt werden kann. Für den Übergang soll sich aus einer vordefinierten Palette an Übergangseffekten wählen lassen.

Der zweite Bereich hängt unmittelbar mit den Mediendateien zusammen. Hier sollen sich Medien jederzeit hinzufügen, ändern und löschen lassen. Dazu sollen sich Optionen, wie die Anzeigedauer, individuell einstellen lassen. Auf die Abspielliste bezogen soll auch die Anzeigereihenfolge für den automatischen Modus frei veränderbar sein und auch jeweils bestimmt

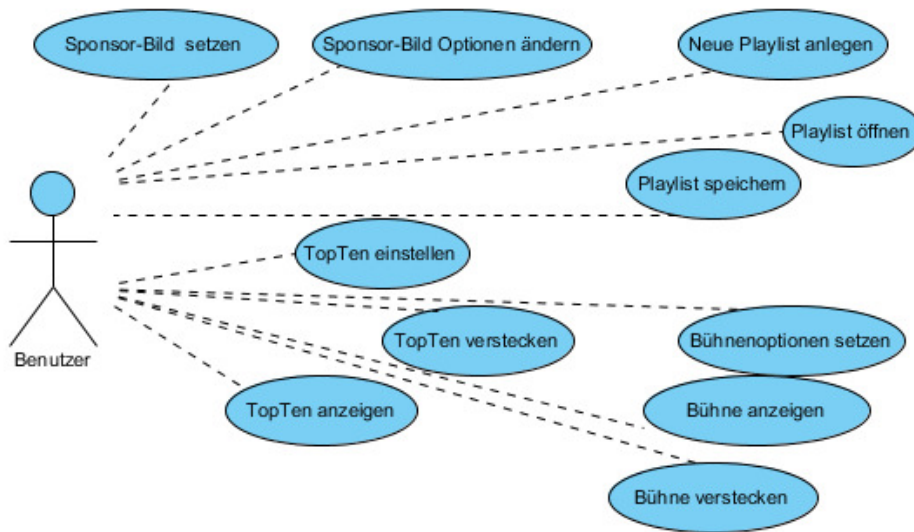


Abbildung 6.4: Nebenfunktionen einer Videomischer-Software

werden können, wenn ein anderes Medium als nächstes angezeigt werden soll.

Die Nebenfunktionen sind in Abbildung 6.4 dargestellt. Diese Funktionen werden nicht so häufig benötigt und müssen deshalb nicht direkt auf der Benutzeroberfläche zugänglich sein. Der erste Bereich umfasst das Sponsor-Bild, welches grundsätzlich einmal eingestellt wird und während einer Veranstaltung gleich bleibt.

Die Top Ten-Anzeige wird als zusätzliches Modul entwickelt und auch nicht auf jeder Veranstaltung benötigt werden. Wenn in einem späteren Entwicklungsstand auch Anzeigen für Spielstände in verschiedenen Sportarten vorhanden sind, kann diese Funktion aber durchaus auch als eine Hauptfunktion angesehen werden.

Über die Bühnenoptionen wird das Ausgabebild eingestellt. Videowände greifen den Monitor unterschiedlich ab, hier soll der Anwender flexibel sein und die Größe und Position des Anzeigebereichs flexibel einstellen können. Diese Einstellung wird in der Regel auch nur einmal vor der Veranstaltung eingestellt und bleibt dann gleich.

Der letzte Bereich sind Playlists. Um die Bedienung zusätzlich zu vereinfachen, soll sich die Liste der abzuspielenden Medien auch speichern lassen. So lassen sich vor der Veranstaltung schon grundsätzliche Dateien hinzufügen, die auf jeden Fall benötigt werden. Hier sollen gängige Aktionen wie Neu, Öffnen und Speichern mit eingebunden werden.

6.4 Zusammenfassung

In diesem Kapitel wurden die Anforderungen an die Software unter verschiedenen Aspekten betrachtet und näher analysiert. Der wichtigste Aspekt war die Echtzeit, aus der viele Schlussfolgerungen für die Konzeption und spätere Umsetzung resultieren. Daraufhin wurden

die Nutzergruppen identifiziert und Benutzerkonzepte entwickelt. Als Grundkonzept wurde hier die Umsetzung eines Videomischers auf Software-Ebene entwickelt. Dieser soll sich aber nur auf die notwendigen Funktionen beschränken und dadurch eine übersichtliche Benutzeroberfläche anbieten, ganz nach dem Konzept: „Form follows Function“. Dadurch wird gewährleistet, dass die grundsätzlich verschiedenen Nutzergruppen alle mit der Software zurecht kommen. Unterstützende Konzepte sind einfache Menüstrukturen und die Anzeige von Tipps während der Benutzung. Diese Konzepte sollen die Bedienung eines digitalen Mixers zusätzlich unterstützen.

Daraufhin wurden die Funktionen durch Use Cases zusammengestellt. Diese wurden im Sinne des Konzepts in zwei Kategorien aufgeteilt:

- Hauptfunktionen, die auf der Benutzeroberfläche direkt zugänglich sein müssen.
- Nebenfunktionen, die in Menüs verlagert werden können.

Die Konzepte und Funktionen bieten nun einen guten Rahmen für die Realisierung. Im nächsten Kapitel sollen daher die Konzepte und Funktionen von der technischen Seite aus betrachtet und prototypisch in einer Software umgesetzt werden.

Kapitel 7

Prototypische Realisierung

In diesem Kapitel werden die Konzepte des letzten Kapitels aufgegriffen und im Hinblick auf die Umsetzung näher analysiert. Dazu wird erläutert wie die Konzepte technisch in einem Prototyp realisiert werden können.

Die Entwicklung der technischen Konzepte erfolgt in zwei Schritten. Zuerst werden die allgemeinen Punkte in Hinsicht auf die technische Umsetzung betrachtet und danach die Punkte, die sich auf die Benutzeroberfläche beziehen. Hierzu wird analysiert, welche Komponente für die Benutzeroberfläche benötigt werden und wie dort eine effektive Umsetzung stattfinden kann. Daraufhin wird die geplante Software-Architektur vorgestellt.

7.1 Technische Konzepte

Durch die zuvor entwickelten Benutzerkonzepte können nun technische Konzepte entwickelt werden. Eine der wichtigsten Anforderungen an die skizzierte Software war die Bedienung in Echtzeit. Demnach müssen benutzerseitige Bedienfehler verhindert und die Benutzer in der Bedienung unterstützt werden. Die im Rahmen dieser Arbeit herausgestellten Anforderungen werden im Folgenden aus technischer Sicht näher betrachtet.

Eine Software für Videowände, die auf Bedienung optimiert ist sollte aus mehreren Komponenten bestehen: Eine Komponente, um die Inhalte steuern und kontrollieren zu können und eine Komponente, auf der die Ausgabe stattfindet. Deshalb wird die Programm-Oberfläche auf zwei Bereiche aufgeteilt: Die **Benutzeroberfläche** und die **Bühne**.

Benutzeroberfläche

Über die Benutzeroberfläche kann mit dem System interagiert werden. Hier sollen später alle Use Cases des geplanten Systems ausgeführt werden können. Die Umsetzung und der Entwurf derselben werden im Teilkapitel 7.2 näher betrachtet.

Bühne

Auf der Bühne wird das letztendliche Ausgabebild für die Videowand erzeugt. Alle über die Benutzeroberfläche gemachten Bedienaktionen werden auf der Bühne direkt sichtbar. Um

die Bühne für alle Videowände möglichst am flexibelsten zu halten, muss sie sich in Größe und Position beliebig einstellen lassen.

Ein wichtiges Konzept für diese geteilte Architektur ist die Vermeidung von kritischen Bedienfehlern, die die Ausgabe auf der Bühne beeinflussen können.

7.1.1 Bedienfehler bei der Echtzeitbedienung

Durch die Ansteuerung der Videowand über den erweiterten Monitor ergibt sich ein großes Fehlerpotenzial. Der Benutzer kann die Bühne auf dem zweiten Monitor recht schnell mit anderen Inhalten überdecken. Es können somit andere Inhalte auf der Bühne auftauchen, als dafür vorgesehen. Dieser Benutzerfehler soll technisch verhindert werden, indem die Bühnenfläche immer im Vordergrund angezeigt wird. Dieses Konzept kann über die Programmierung umgesetzt werden. Somit bleibt die Bühne immer von Interaktionen außerhalb des Programms unberührt.

Dieser Fehler war ein Benutzerfehler, der jedoch wenig mit der eigentlichen Bedienung des Systems zu tun hat. Durch die Bedienung können die meisten anderen Fehler auftreten. Das Vertrauen der Anwender sollte insgesamt durch eine gute Funktionalität und Fehlerrobustheit gewährleistet werden. Fehlerrobustheit bedeutet technisch gesehen, dass bei einer Echtzeitsoftware keine kritischen Fehler auftreten, die die Ausgabe beeinflussen. Da der Benutzer sehr flexibel bei der Verwaltung von Medien sein soll und diese jederzeit hinzufügen, löschen und umsordieren kann, muss die Software die gewünschten Aktionen beachten und auch dementsprechend verarbeiten können. Wenn beispielsweise das Medienobjekt, das als nächstes angezeigt werden soll, gelöscht wird, darf kein Fehler produziert werden, sondern die Software sollte mit dem nächsten Objekt fortfahren. Auch andere Programmfehler sollten diskret auftreten und die Präsentation auf der Bühne so wenig wie möglich beeinflussen.

Weitere Konzepte mit dem Schwerpunkt auf Echtzeitbedienung beziehen sich unmittelbar auf die Benutzeroberfläche und werden im Teilkapitel 7.2 entwickelt.

Andere Anforderungen werden im Folgenden näher betrachtet und parallel dazu mögliche Lösungen entwickelt.

7.1.2 Allgemeine Konzepte

Üblicherweise werden für eine Veranstaltung im Voraus schon Mediendateien gesammelt und zum Testen in die Software eingefügt. Damit diese nicht nochmal auf der Veranstaltung, wo meistens auch wenig Zeit vorhanden ist, neu eingeladen werden müssen, wird deshalb eine Möglichkeit entwickelt, eingeladene Medien als **Playlist** abspeichern zu können. Diese Playlist kann später auf der Veranstaltung einfach geladen und in Nachhinein noch bearbeitet werden. Dadurch wird eine hohe Flexibilität erreicht.

Für die Anzeige von Spielständen und Ergebnissen wird eine weitere Umsetzungsstrategie benötigt. Die Daten müssen von einer externen Datenquelle in das System eingespeist werden. Hierzu hilft die Unterstützung von Webservices in Flash weiter. Webservices sind

Software-Anwendungen, die eine Schnittstelle für den Austausch verschiedener Daten bereitstellen. Demnach kann im späteren Netzwerk ein Webservice zur Verfügung gestellt werden, der Ergebnisdaten und Spielstände bereitstellt. Über Flash können diese dann abgegriffen und formatiert werden.

Ein weiteres K.O.-Kriterium ist die Anzeige von Live-Videos und Videostreams. Diese werden, wie bereits genannt, über zwei Arten eingespielt. Live-Videos können sehr einfach von einer Videokarte abgegriffen werden. Der Zugriff wird von Flash gekapselt und ist über die Software leicht zugänglich. Videostreams hingegen können sehr viele verschiedene Formate haben. Neben vielen proprietären Vertretern gibt es auch freie Formate, die weit verbreitet sind. In Kombination mit Netzwerk-Kameras werden Motion-JPEG (MJPEG)-Streams sehr häufig verwendet. Diese bestehen aus einer Abfolge von einzelnen JPEG-komprimierten Bildern. In dem Prototyp soll erstmal für diese Möglichkeit eine Lösung entwickelt werden.

Flash unterstützt eine Vielfalt an Medienformaten, die deshalb auch in der Software angeboten werden sollen. Diese werden hier nur kurz, der Vollständigkeit halber, aufgezählt:

- Bilder (JPEG, GIF, PNG)
- Videos (F4V, FLV, MOV, MP4)
- Flash (SWF)
- Websites (HTM, HTML)
- Rich-Text über direkte Eingabe von HTML-Code








Die noch nicht betrachteten Anforderungen beziehen sich auf die Bedienung des eigentlichen Systems. Eines der wichtigsten Konzepte ist hier die Entwicklung einer effektiven und effizienten Benutzeroberfläche.


7.2 Benutzeroberfläche

Eine der wichtigsten Punkte der Mensch-Computer-Interaktion, neben der Funktionalität, ist die Benutzeroberfläche. Über sie interagiert der Benutzer mit dem Computer und tauscht Informationen aus. Die meisten der im vorherigen Kapitel herausgestellten Anforderungen, beziehen sich unmittelbar auf die Benutzeroberfläche. Um eine Benutzerschnittstelle für den Menschen nutzbar zu gestalten, muss sie auf die Bedürfnisse des Benutzers angepasst sein. Aus der Überlegung, welche Komponenten für die Benutzeroberfläche benötigt werden, lassen sich direkt drei wichtige Komponenten feststellen:

1. Bus
2. Fadebar
3. Vorschaufenster

7. PROTOTYPISCHE REALISIERUNG

	Name	Aktiviert	Anzeigedauer
	IMG_1135	<input checked="" type="checkbox"/>	6
	IMG_1136	<input checked="" type="checkbox"/>	6
	IMG_1137	<input checked="" type="checkbox"/>	6
	IMG_1138	<input checked="" type="checkbox"/>	6
	thelegendofzeldamusik	<input checked="" type="checkbox"/>	0
	chipzeit_logo_drehen_480320	<input checked="" type="checkbox"/>	10
	HeyHo	<input checked="" type="checkbox"/>	15

 Medium Hinzufügen

- Aktuell ausgewählt
- Aktives Medium
- Vom Benutzer als nächstes Medium gewählt

Abbildung 7.1: Der Bus wird durch ein tabellenartiges Steuerelement (DataGrid) abgebildet und kann somit auch direkt wichtige Informationen zu den Medienobjekten anzeigen.

Bus

Da Mediendateien flexibel hinzugefügt und entfernt werden müssen, stellen Buttons für den Bus keine praktische Umsetzungsmöglichkeit dar. Es wäre für den technisch Versierten zwar leicht erkennbar, dass es sich um eine Art Bus handelt, jedoch wäre die Bedienung auch ziemlich ungewöhnlich. Wenn neue Medienobjekte hinzugefügt werden sollen, müsste pro Medium ein neuer Button erstellt werden. Eigenschaften zu den Objekten könnten zudem nur über die Buttons (beispielsweise über einen Rechtsklick) verfügbar sein. Dieses Verhalten ist in gängiger Software eher ungewöhnlich, deshalb sollte sich hier von Buttons gelöst und ein anderes Bedienkonzept entwickelt werden.

Das Ziel ist allerdings auch nicht, eine Kopie des Videomischers zu entwickeln, sondern eine gute Alternative, die funktionierende Konzepte des Videomischers auf eine Software überträgt. Zur Auflistung vieler Medienobjekte wird hier deshalb eine Tabelle vorgezogen, in der die einzelnen Einträge untereinander angezeigt werden.

Eine Tabelle bietet viele verschiedene Möglichkeiten der Bedienung an. Objekte können beispielsweise einfach per Drag and Drop umsortiert werden. Ansonsten können auch viele Funktionen des sogenannten DataGrids von Flash genutzt werden. Hier können die Eigenschaften der einzelnen Medienobjekten in Spalten angezeigt werden. Die erste Spalte zeigt durch ein Symbol an, um welchen Dateityp es sich bei dem jeweiligen Objekt handelt. Durch die Auswahlboxen in der dritten Spalte lassen sich Objekte aus dem automatischen Modus herausnehmen und werden dann nicht mehr angezeigt. Über das Kontextmenü kann ein Eigenschaftenfenster für jedes Objekt aufgerufen und jeweils eine Anzeigedauer festgelegt werden. Diese wird auch in der Liste angezeigt. Man sieht dass durch die Auflistung der Medienobjekte in einem DataGrid sehr viel mehr Informationen angezeigt werden können, als in Kombination mit Buttons. Somit bietet diese Lösung hier eine viel bessere Funktionalität. Dazu sollte das Tabellensystem für die Benutzer intuitiv anwendbar sein, da es aus anderen Programmen bekannt sein sollte.

Ein Punkt bezogen auf die Übersichtlichkeit, muss bei der Tabelle im Vergleich zum Bus

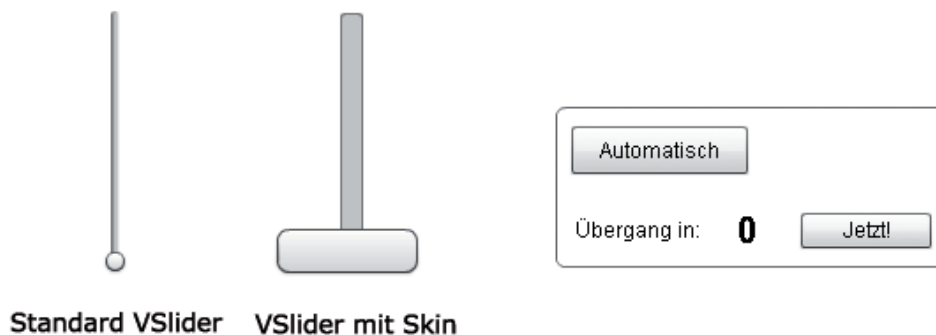


Abbildung 7.2: Die Fadebar kann aus einem VSlider-Steuerelement durch Skins in Flash entwickelt werden. Die Präsentation kann auch automatisch ablaufen (siehe Abbildung rechts). Hierzu wird ein Countdown angezeigt, wann der nächste Wechsel erfolgt. Ein unmittelbarer Wechsel kann durch Klick auch erzwungen werden.

jedoch anders gehandhabt werden. Bei einem Videomischer sieht man durch die gedrückten Knöpfe auf dem Bus, welcher Eingang aktuell gewählt ist. Bei einer Tabelle ist das so nicht möglich. Für die Umsetzung in dem Prototyp werden deshalb verschiedene Farben und Symbole eingeführt. Das aktive Objekt wird hier durch eine spezielle Farbe hinterlegt. Das nächste Objekt ist intuitiv auch das nächste in der Liste. Wenn ein anderes Objekt als nächstes angezeigt werden soll, kann es über das Kontextmenü ausgewählt, oder einfach angeklickt werden. Dieses bekommt dann ein kleines Stern-Symbol als Markierung. Der letzte sehr wichtige Punkt ist das Hinzufügen von neuen Medienobjekten zur Präsentation. Hierzu gibt es im Bereich des Bus einen Button, über den Mediendateien hinzugefügt werden können. Dieser ist ein zentrales Steuerelement und wurde deshalb durch Größe und Symbol hervorgehoben (siehe Abbildung 7.1).

Fadebar

Für die Fadebar kann kein einfaches Standard-Steuerelement verwendet werden. Eine ähnliche Komponente im Flex-Framework ist jedoch der VSlider (vertikaler Schieberegler). Durch die Fähigkeit des Skinings kann dieses Element auf die eigenen Bedürfnisse angepasst werden und wird dadurch der Fadebar nachempfunden (siehe Abbildung 7.2). Zusätzlich soll auch ein automatischer Modus eingeführt werden, der Übergänge selbständig durchführen kann. Diese Steuerung bietet einen sehr großen Vorteil gegenüber einem Videomischer, denn die Liste der anzuzeigenden Medien kann hier durchiteriert werden. Die automatische Steuerung soll hier durch einen Countdown unterstützt werden, der anzeigt, wann der Wechsel zum nächsten Medium stattfindet. Zu der Fadebar gehören auch Übergangseffekte, die sowohl im manuellen, als auch im automatischen Modus verwendet werden können. Hier soll einerseits der Effekt und andererseits auch die Übergangsrichtung wählbar sein (siehe Abbildung 7.3).

Alle diese Funktionen werden in einem Bereich untergebracht — dem Übergangsbereich.

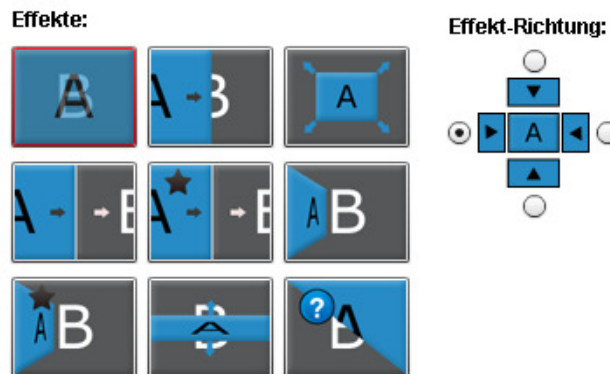


Abbildung 7.3: Die Effekte werden durch Vorschaubilder verdeutlicht, auch die Effekt-Richtung kann gewählt werden.

Vorschaufenster

Das Vorschaufenster wird über ein Bild-Feld abgedeckt, dass sich alle 40 Millisekunden aktualisiert. Das entspricht einer Bildwiederholfrequenz von 25 Bildern pro Sekunde und reicht somit für eine flüssige Darstellung aus. Zusätzlich wird nicht nur das aktuelle Medienobjekt, sondern auch ein Vorschau vom nächsten Medienobjekt dargestellt. Hier bekommt der Benutzer eine gute Übersicht, was sich gerade auf der Videowand abspielt und als nächstes abspielen wird. Das ist durchaus sinnvoll, denn nicht immer hat der Benutzer der Software einen direkten Blick auf die Videowand. Besonders bei einer Bedienung in Echtzeit muss dem Benutzer eine Sicherheit gegeben werden, um möglichst wenig Fehler zu begehen. Durch die Vorschaufenster bekommt er somit eine Rückmeldung und diese Sicherheit ist gewährleistet.

Durch diese drei Gruppierungen der Komponenten werden die meisten der Hauptfunktio-

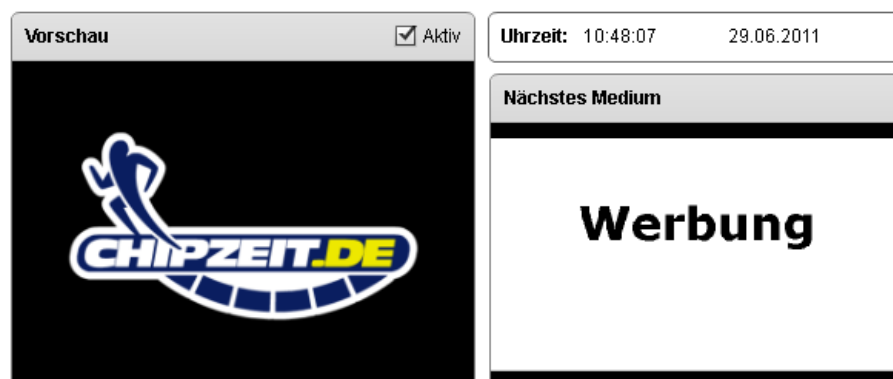


Abbildung 7.4: Das linke Fenster zeigt das aktuell angezeigte Objekt, im kleineren rechten Fenster wird das nächste Objekt angezeigt.

nen abgedeckt. Die Präsentation lässt sich durch die Fadebar beliebig steuern und auch ein automatischer Modus ist bei den Übergangselementen enthalten. Auch können die Medienobjekte durch den neu entwickelten Bus, in Form einer Tabelle, verwaltet werden. Die Anzeigereihenfolge lässt sich durch Umsortieren über Drag and Drop jederzeit ändern. Hier wurde ein Kontextmenü entwickelt, das per Rechtsklick auf den jeweiligen Eintrag in der Tabelle geöffnet wird. Über dieses Menü können die Optionen des Objekts jederzeit geändert und das Objekt auch gelöscht werden.

Des Weiteren wurden noch Nebenfunktionen genannt, die in der Software enthalten sein sollen. Gegenüber den Hauptfunktionen, die über einen Klick erreichbar sind, werden Nebenfunktionen deshalb in Menüs verlagert. Dies hat mehrere Gründe:

- Viele Funktionen werden nur selten benötigt, beispielsweise um initiale Einstellungen zu treffen.
- Andere Funktionen sind sehr speziell und werden nur auf manchen Veranstaltungen genutzt, ein Beispiel sind Spielstandsanzeigen.
- Das Design der Benutzeroberfläche wirkt dadurch aufgeräumter und die Hauptfunktionen erhalten dadurch mehr Aufmerksamkeit und rücken mehr in den Vordergrund.

Im Folgenden werden die Nebenfunktionen genannt und erläutert, wie diese zugänglich gemacht werden sollen.

Die Nebenfunktionen werden, wie bereits genannt, in die Menüleiste der Software integriert. Diese teilt sich in drei Bereiche auf: **Datei**, **Bühne** und **Modul**.

Datei

Über das Dateimenü können die bereits erwähnten Playlists erstellt, geöffnet und gespeichert werden. Das funktioniert analog, wie bei vielen anderen Programmen auch und sollte dem Benutzer deshalb dementsprechend intuitiv vorkommen. Als nützliches Feature wurde hier eine Option hinzugefügt, um zuletzt verwendete Playlists direkt öffnen zu können.

Bühne

Über dieses Menü können zum einen Bühnenoptionen, wie beispielsweise Position und Größe angepasst werden. Als Schnellzugriff steht hier im Menü direkt die Möglichkeit bereit, die Bühne verstecken oder wieder anzeigen zu lassen. Zum anderen kann hier auch ein Sponsor-Bild geladen werden, welches dann rechts oben auf der Bühne angezeigt wird. Bei den Einstellungsmenüs wurde direkt darauf geachtet, benutzerseitige Fehler zu verhindern. Somit wurden Steuerelemente verwendet, welche Dateneingaben nur so zulassen, wie sie benötigt werden.

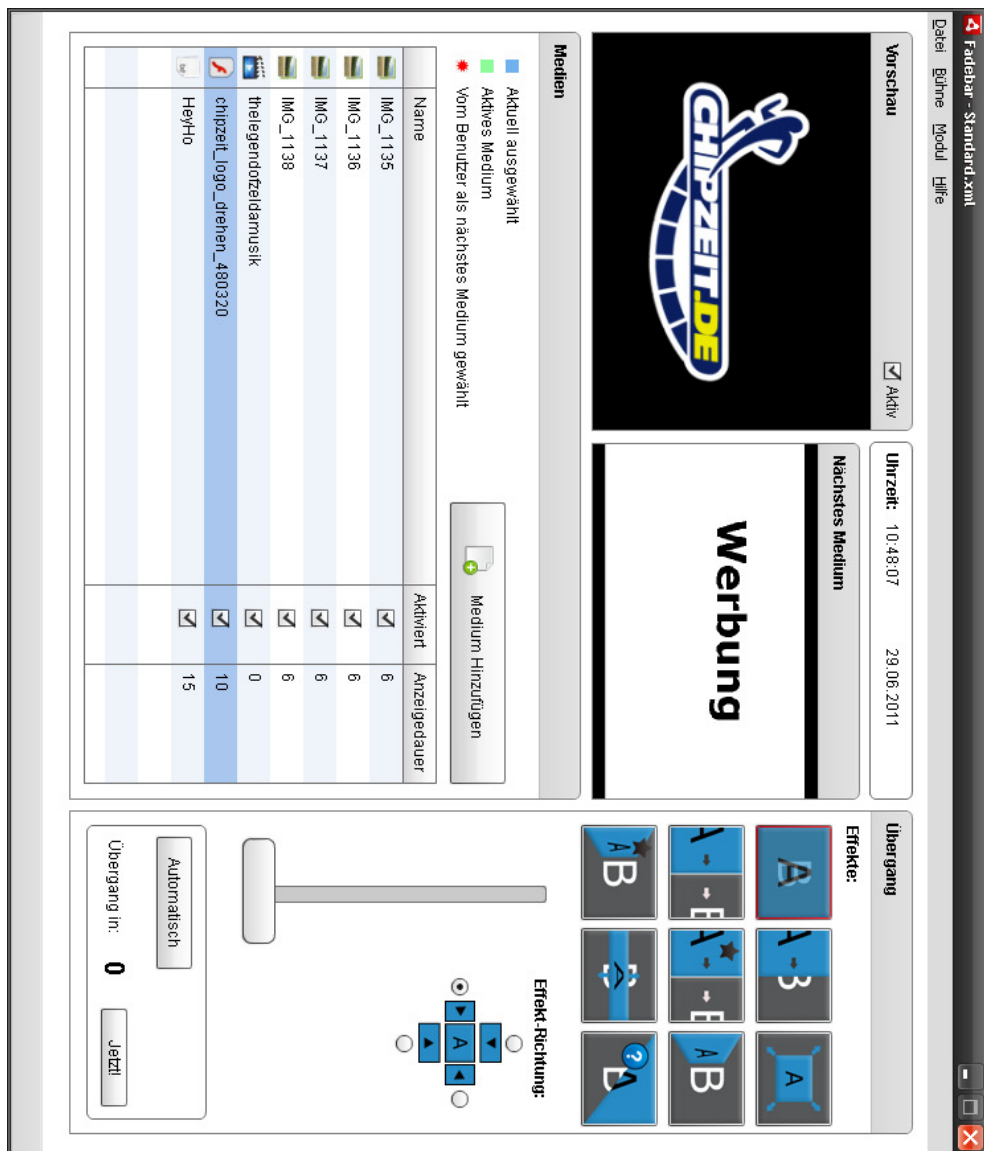


Abbildung 7.5: Der Gesamtentwurf des Prototypen, bestehend aus den einzelnen Komponentengruppen (Vorschaubereich, Medienliste, Fadobar-Bereich).

Modul

Im Bereich Modul gibt es die Möglichkeit, auf optionale Softwaremodule zurück zu greifen. Für den Prototyp gibt es hier die Möglichkeit Top Ten-Ergebnisse von einem Webservice abzurufen und dann auf der Bühne anzuzeigen. Die Ergebnisse werden hierbei halbtransparent über den anderen Inhalten angezeigt.



Abbildung 7.6: Die IMediaObject-Schnittstelle sorgt dafür, dass bei den verschiedenen Medienobjekten gleiches Verhalten implementiert wird.

7.3 Architektur

Ein entscheidender Punkt bei einer Präsentationssoftware sind die vielen verschiedenen Medientypen. Diese verhalten sich alle unterschiedlich, sollen jedoch alle gleichartig in einer Präsentation verwendet und angesprochen werden können. Deshalb muss jeder dieser Typen weiterentwickelt werden, um ein gleiches Verhalten zu erreichen. Das Ganze wird durch eine Schnittstelle (Interface) umgesetzt, die alle Medientypen realisieren. In Abbildung 7.6 sind die benötigten Methoden zu sehen. Hauptsächlich handelt es sich dabei um Set- und Get-Methoden, um verschiedene Werte zu belegen und abzufragen. Durch diesen Aufbau lassen sich alle Medienobjekte gleich ansprechen. Folgende Eigenschaften sind entsprechend der entwickelten Architektur typisch für ein Medienobjekt:

- **ObjectName** — Durch den Namen kann ein Objekt in der Anzeigeliste identifiziert werden.
- **Anzeigedauer** — Gibt für den automatischen Modus an, wie lange ein Objekt jeweils angezeigt werden soll.
- **DisplayImage** — Stellt ein kleines Symbol dar, das in der Anzeigeliste für das Objekt angezeigt wird. Hierzu wurden pro Medientyp einzelne Symbole entwickelt. Es wäre jedoch durchaus in einer weiteren Ausbaustufe möglich, jedes Objekt mit einem individuellen Bild zu versehen.

- **PlayEnabled** — Ist dieser Wert aktiviert, so wird das Objekt im automatischen Modus auch angezeigt.
- **ActiveOnStage** — Diese Option wird durch die Steuerklasse verwaltet und gibt an, ob sich ein Objekt derzeit auf der Bühne befindet.
- **ShowNext** — Durch diese Option wird festgelegt, dass ein Objekt als nächstes angezeigt werden soll.
- **Initialized** — In Flash muss jedes Anzeigeobjekt erstmal der Bühne hinzugefügt werden bevor es seine Eigenschaften wie beispielsweise die Größe preisgibt. Diese Eigenschaft gibt deshalb an, ob die Werte schon zur Verfügung stehen.
- **MediaScaleMode** — Diese Einstellung entscheidet, ob ein Medienobjekt auf seine Höhe oder Breite skaliert wird.

Aus diesem Konzept heraus lässt sich die letztendliche Architektur besser nachvollziehen. Wie bereits genannt, wird der Prototyp in Actionscript 3 und MXML entwickelt. Dementsprechend ist das Projekt auch objektorientiert. In folgendem Klassendiagramm ist die grundsätzliche Architektur mit den wichtigsten Komponenten dargelegt (siehe Abbildung 7.7).

Fadebar

Die Hauptkomponente ist die Klasse **Fadebar** — diese erzeugt die Benutzeroberfläche und verwaltet die dort stattfindenden Interaktionen. Die Benutzeroberfläche wird komplett durch MXML beschrieben und erzeugt eine Oberfläche, wie in Abbildung 7.5 zu sehen. Von dieser Komponente werden auch die Playlists verwaltet (im Diagramm als **Template** bezeichnet).

Template

Die **Template**-Klasse kümmert sich um das Laden und Speichern der Playlists. Die Playlists werden als XML-Dateien gespeichert. Diese Funktionalität soll den Benutzer zusätzliche Arbeit abnehmen.

MediaManager

Eine der wichtigsten Klassen, der **MediaManager**, wird auch von der Hauptkomponente erzeugt. Diese Klasse steuert und verwaltet die Präsentation. Hierzu enthält es eine Liste aller in der Präsentation enthaltenen Medienobjekte (**IMediaObject-Collection**). Ein Auszug der möglichen Medienobjekte befindet sich im unteren Teil des Diagramms. Der **MediaManager** kümmert sich um die Auswahl des nächsten Mediums und auch darum, dass es der Bühne zeitnah übergeben wird. Funktionen wie der automatische Abspielmodus oder Änderung der Bühnenoptionen werden auch von dieser Klasse verwaltet.

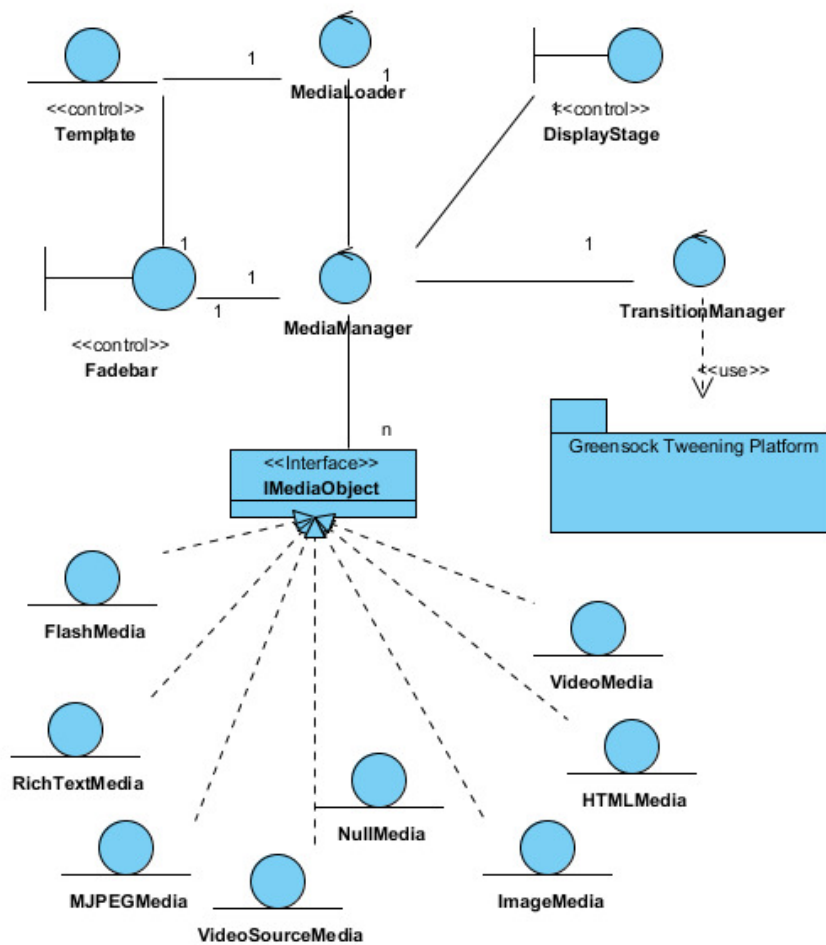


Abbildung 7.7: Auszug vom Klassendiagramm des Prototyps mit den wichtigsten Klassen der Architektur.

MediaLoader

Flash hat die Besonderheit, Optionen zu Objekten erst bereitzustellen, wenn diese bereits einmal einer Bühne hinzugefügt worden sind. Deshalb muss jedes Medienobjekt vor seiner ersten Verwendung initialisiert werden. Hierfür ist der MediaLoader verantwortlich. Dieser kapselt im ersten Schritt die verschiedenen Medientypen in den jeweiligen Objekten und führt dann eine Initialisierung durch. Durch diese wird das Objekt dann auch in die, für die Bühne richtige Größe gebracht. Je nach Skalierungsmodus des Objekts, wird die Größe der Höhe oder Breite angepasst.

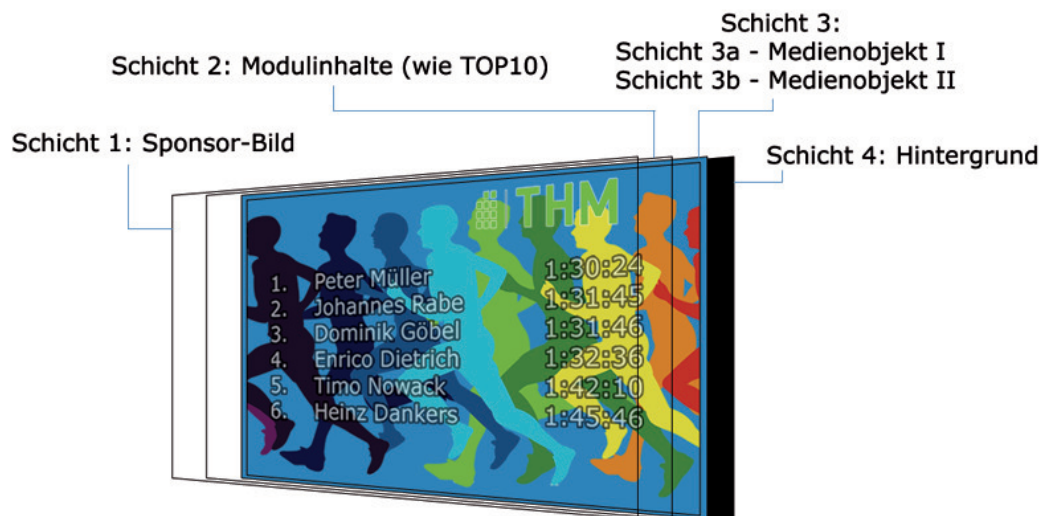


Abbildung 7.8: Die verschiedenen Anzeigeschichten der Bühne.

DisplayStage

Die DisplayStage ist die Bühne, auf der die Bildausgabe stattfindet. Sie besteht aus vier verschiedenen Anzeigeschichten, auf die jeweils Inhalte gebracht werden können:

1. Auf der obersten Schicht ist das Sponsor-Bild enthalten, das sich somit auch immer im Vordergrund befindet.
2. Diese Schicht ist für Modulausgaben reserviert. Beispielsweise finden hier Ausgaben des Top Ten-Moduls statt.
3. Hier befindet sich die Master-Schicht. Auf dieser werden die eigentlichen Medienobjekte angezeigt. Um Übergangseffekte zwischen zwei Objekten durchführen zu können, besteht diese Schicht wiederum aus zwei einzelnen Schichten. Bei einem Übergang, wird die eine Schicht ausgeblendet und die andere eingeblendet. Daraufhin wird die inaktive Schicht in den Hintergrund gerückt und darauf das nächste Medienobjekt geladen.
4. Die unterste Schicht besteht lediglich aus einem Rechteck, das die Hintergrundfarbe erzeugt.

TransitionManager

Diese Klasse kann Übergänge zwischen zwei verschiedenen Medienobjekten erzeugen. Hierzu greift es auf die Greensock Tweening Plattform zurück, die im Grundlagenkapitel schon erwähnt wurde. Für den Prototyp wurden 8 verschiedene Übergangsarten entwickelt, die

verwendet werden können. Diese bestehen hauptsächlich aus Überblendungs- und Wisch-Effekten. Die einzelnen Effekte sind mit der Fadebar auf der Benutzeroberfläche verbunden und können über diese gesteuert werden.

7.4 Zusammenfassung

In diesem Kapitel wurden technische Konzepte entwickelt, welche die Usability in der Software unterstützen sollen. Besonders wichtig war hierbei die Entwicklung der Benutzeroberfläche für den Prototyp. Diese ist zum Steuern der Multimedialinhalte auf der Videowand vorgesehen und von der Präsentationsschicht getrennt. Die Bedienoberfläche ist sehr aufgeräumt und besteht aus drei grundsätzlichen Bereichen: Der Vorschau, der Medienliste und dem Übergangs-Bereich.

Die Präsentationsebene oder auch Bühne, ist in 4 (5) Schichten aufgeteilt. Hierdurch ist gewährleistet, dass alle Inhalte getrennt voneinander auf die Videowand gebracht werden können und eine gute Flexibilität gewahrt wird.

Im nächsten Schritt wurde die Architektur des Prototyps, der die vorher entwickelten Konzepte umsetzt, näher beschrieben. Alle Medienobjekte werden durch eine Schnittstelle, dem IMediaObject, zusammengefasst. Die wichtigsten Klassen verwalten diese einzelnen Medienobjekte und sorgen für die Anzeige auf dem Bildschirmbereich der Videowand.

Ob die Zielvorgaben erreicht wurden, wird im nächsten Kapitel durch die Ergebnisse und eine Evaluation näher untersucht.

Kapitel 8

Ergebnisse

In diesem Kapitel werden die im Rahmen dieser Arbeit entwickelten Ergebnisse präsentiert. Hierzu wird gezeigt, wie die eingangs gestellten Probleme durch die verwendete Methodik und den entwickelten Prototypen gelöst werden. Dafür wird die entwickelte Lösung in Bezug auf den aktuellen Stand der Technik und existierende Software betrachtet. Daraufhin werden die Ergebnisse über eine durchgeführte Evaluation verifiziert und hier aufgetretene Schwachpunkte noch mal näher betrachtet.

8.1 Entwickelte Konzepte

Durch die Untersuchungen hat sich herausgestellt, dass eine Bedienung in Echtzeit für Sportveranstaltungen eine große Rolle spielt. Hierfür eigneten sich die Bedienkonzepte eines Videomischers sehr gut, da dieser bereits eine Bedienung von Videoinhalten in Echtzeit möglich macht. Deshalb wurden diese Konzepte auf die gestellten Anforderungen angepasst und daraufhin umgesetzt. Die Software macht nun die Präsentation von Multimediainhalten auf einer Videowand in Echtzeit möglich. Hierzu wurde die Software in zwei Komponenten aufgeteilt: **Die Benutzeroberfläche**, auf der die Medienobjekte gesteuert und verwaltet werden können. **Die Bühne**, auf der die Präsentation der Inhalte stattfindet. Durch diese Aufteilung lassen sich die Inhalte flexibel in Echtzeit steuern und der Benutzer hat jederzeit die volle Kontrolle über die Präsentation. Es hat sich hier zudem herausgestellt, dass Bedien- und Programmfehler ein ernst zu nehmender Faktor sind. Diese müssen daher vermieden werden, eine gute Usability ist daher insbesondere wichtig. Die Bedienung in Echtzeit sollte zudem durch das Anzeigen von Hilfestellungen unterstützt werden, damit der Benutzer sich auch bei nicht so häufig verwendeten Funktionen sicher fühlt.

Ansonsten ist eine Analyse der Programmfunktionen wichtig. Es sollten nur Bedienelemente der Hauptfunktionalitäten auf der Programmoberfläche direkt zugänglich gemacht werden. Andere Elemente sollten in Menüs verlagert werden. Dadurch soll eine höhere Übersichtlichkeit bei der Bedienung ermöglicht werden, die sich bei einer Echtzeitbedienung als besonders wichtig herausgestellt hat.

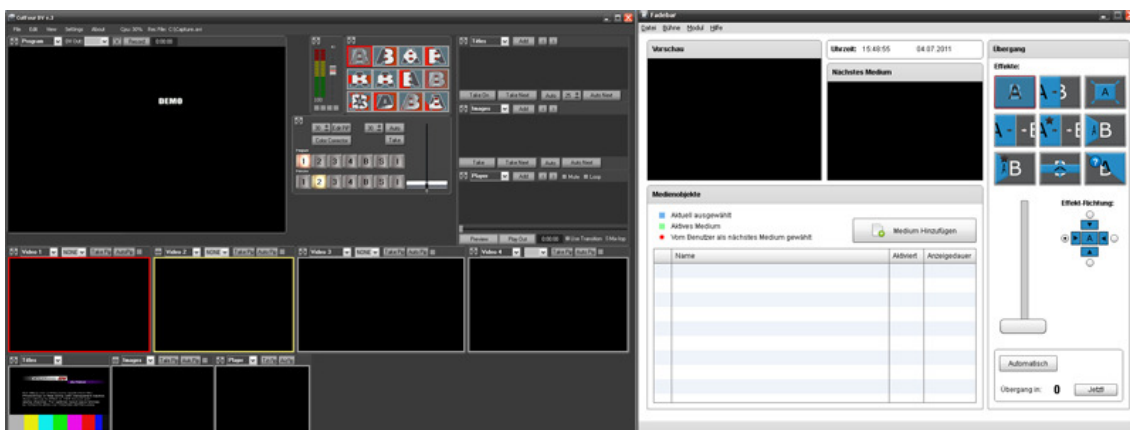


Abbildung 8.1: Vergleich der Benutzeroberfläche eines Playout-Systems (links) und dem entwickelten Prototypen (rechts). Der Prototyp wirkt sichtlich aufgeräumter und verfügt über deutlich weniger Steuerelemente.

8.2 Betrachtung des Prototypen

Die Anforderungen an eine Echtzeitsoftware im Sport- und Multimedia-Bereich wurden in dem entwickeltem Prototyp umgesetzt. Auf der Benutzeroberfläche wird eine Bedienung in Echtzeit durch die Verwaltung der einzelnen Medienobjekte in einer Liste unterstützt. Die hinzugefügten Mediendateien werden hier untereinander angezeigt. Es ist jederzeit möglich neue Mediendateien hinzuzufügen oder bestehende zu löschen. Auch kann während der Präsentation zu jedem beliebigen Objekt gewechselt werden. Das kann manuell über die Fadebar oder automatisch und zeitgesteuert erfolgen. Hierbei können Mediendateien sehr vieler Dateitypen verwendet und angezeigt werden. Die ganze Bedienung ist in Echtzeit möglich, die Software setzt demnach Bedienaktionen direkt auf der Videowand um. Unterstützt wird die Bedienung durch zwei Vorschaufenster, die dem Benutzer zeigen, was sich auf der Videowand abspielt und welches Objekt als nächstes angezeigt wird. Das zweite Vorschaufenster unterstützt den Benutzer zusätzlich, weil er dadurch überprüfen kann, dass auch die richtigen Inhalte als nächstes auf die Videowand gebracht werden. Diese Funktion ist insbesondere nützlich, um Fehler bei der Bedienung zu vermeiden, die sich besonders kritisch auswirken können. Für die Bühne wurde eine 4/(5)-Schichten Architektur entwickelt, welche die verschiedenen Inhalte auf der Videowand voneinander getrennt anordnet und für die nötige Flexibilität sorgt.

Im Sportbereich lässt sich die Software gut einsetzen, weil einerseits die Echtzeitbedienung dort eine große Rolle spielt. Andererseits weil die Möglichkeit gegeben ist, über Module diverse Anzeigen zur Sportveranstaltung einzurichten, wie Top Ten-Ergebnisse. Daher ist der Prototyp nicht auf eine Sportart festgelegt, sondern kann auf Sportveranstaltungen aller Arten gleichwertig eingesetzt werden.

Schon in den Bedienkonzepten unterscheidet sich der Prototyp maßgeblich von bisheriger Software. Betrachtet man die Art der Bedienung, sind die ähnlichsten Anwendungen

die Playout-Systeme, die den Videomischer teilweise eins zu eins umsetzen. Jedoch werden durch den Prototyp noch mehr Freiheiten in der Bedienung und eine höhere Flexibilität durch die Medienvielfalt gegeben. Diese werden jedoch nicht durch eine Verringerung der Übersichtlichkeit ermöglicht, sondern wurden auf einer einfachen Benutzeroberfläche umgesetzt. Diese ist somit auf wesentliche Steuerelemente beschränkt und bietet nur Funktionen an, die am häufigsten benötigt werden. Der Fokus bei der Entwicklung lag hier bei der Gebrauchstauglichkeit. Durch diesen Punkt hebt sich der Prototyp ebenfalls von Playout-Systemen ab, die oft eine Unübersichtlichkeit über die mit Funktionen überhäufte Benutzeroberfläche erzeugen. Wissenschaftliche Untersuchungen im Bereich der Mensch-Computer-Interaktion waren für die Entwicklung sehr hilfreich. Auch die Grundsätze der Dialoggestaltung haben einen wesentlichen Teil zur Umsetzung beigetragen. Ansonsten wurden die eigens entwickelten Benutzerkonzepte verwendet und umgesetzt.

Die erzielten Ergebnisse lassen sich durch eine Evaluation nochmals überprüfen. Der Versuchsaufbau und die Ergebnisse werden im nächsten Teilkapitel beschrieben.

8.3 Evaluation

Als weiteres Ergebnis wurde eine Evaluation des Prototypen durchgeführt, um die Resultate zusätzlich überprüfen zu können. Für die Evaluation wurde ein Benutzertest mit anschließendem Interview und Fragebogen genutzt. Basis hierfür waren 7 Standardaufgaben, die jede Testperson durchgeführt hat. Vor und nach den Aufgaben hatte jede der Testpersonen Zeit sich mit der Software auseinanderzusetzen und sich somit auf den Fragebogen und das darauf folgende Interview vorzubereiten.

8.3.1 Versuchsaufgaben

Als Versuchsaufbau diente ein Computer mit einem zweiten Monitor, der die Videowand darstellen sollte. Die Testpersonen bekamen vor der Versuchsdurchführung eine kurze Schulung wofür die Software verwendet wird und eine grundsätzliche Erklärung der Funktionen.

Daraufhin wurden folgende Aufgaben von den Testpersonen durchgeführt:

1. Richten Sie die Videowand-Ausgabe ein. Stellen Sie hierzu eine Auflösung der Bühne von Breite:400 Höhe:360 bei der Position X:1680 Y:0 ein.
2. Stellen Sie ein Sponsoren-Icon auf der Videowand ein. Dieses befindet sich auf dem USB-Stick unter Medien/Sponsor — Stellen Sie hierzu eine Transparenz von 40% ein.
3. Fügen sie der Präsentation folgende Medien aus dem Ordner Medien/Alle hinzu: Chipzeit.swf, Becks.png, McDonalds.jpg, Demo.png. Lassen Sie zunächst die Flashdatei Chipzeit auf der Videowand anzeigen.
4. Fügen Sie jetzt noch die HTML-Datei info.html aus dem Ordner Medien/Alle hinzu und erzeugen einen manuellen Übergang mit dem Schieberegler zu diesem neuen Objekt. Verwenden Sie hierzu einen beliebigen Wischübergang von oben nach unten.

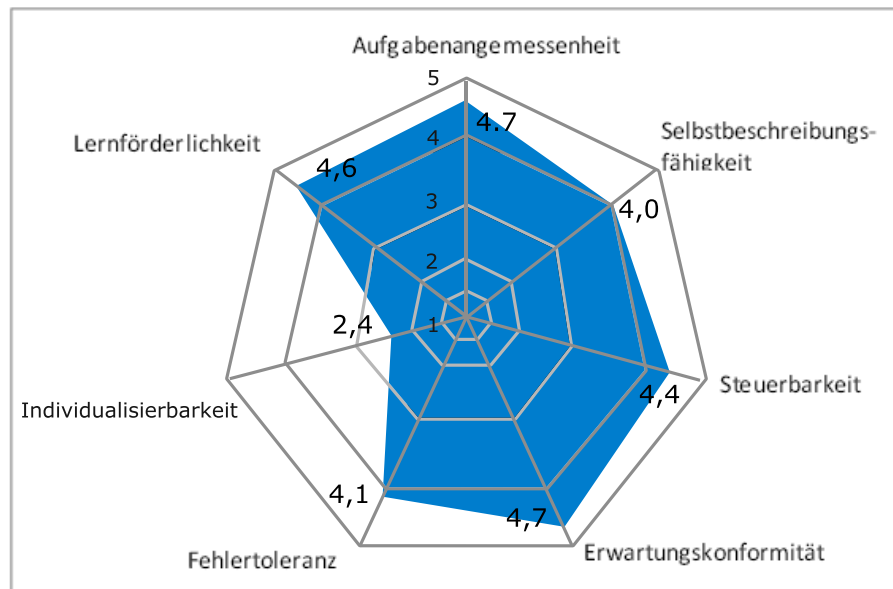


Abbildung 8.2: Beurteilung der Usability durch den Fragebogen (Mittelwerte).

5. Stellen Sie die Anzeigedauer von McDonalds.jpg auf 10 Sekunden ein und sortieren die Medien in folgende Reihenfolge: 1-McDonalds.jpg, 2-Becks.png, 3-Demo.png, 4-Chipzeit.swf, 5-info.html. Danach starten Sie den automatischen Modus und löschen das Medium info.html.
6. Nur wenn eine Internetverbindung besteht: Fügen Sie während der Präsentation noch einen Videostream hinzu. Die Stream-Adresse lautet: 213.196.182.244/mjpg/video.mjpg (Auflösung des Streams: 352x288). Lassen Sie den Stream als nächstes Objekt anzeigen.
7. Speichern Sie die Präsentation als Vorlage auf dem USB-Stick unter Medien/Vorlage ab.

Diese Aufgaben decken einige Gebiete ab, die durch vorhandene Software nicht gelöst werden können und erst durch den Prototyp möglich gemacht werden. Diese sind so konzipiert, dass sie auch in realen Szenarien auftreten könnten.

Die Benutzerbefragung wurde mit dem Standardfragebogen IsoMetrics¹ durchgeführt. Dieser deckt eine Untersuchung der Software auf die 7 Grundsätze der Dialoggestaltung ab.

8.3.2 Ergebnisse der Evaluation

Die Evaluation wurde mit fünf Benutzern durchgeführt, die mit den späteren Nutzergruppen vergleichbar sind. Die Ergebnisse der Benutzerbefragung sind in Abbildung 8.2 zu sehen. Die

¹<http://www.isometrics.uni-osnabrueck.de>

Grundsätze der Dialoggestaltung sind demnach weitgehend gut umgesetzt worden. Besonders wichtig für den Prototyp waren hier die Steuerbarkeit, Aufgabenangemessenheit und Lernförderlichkeit, die auch sehr gute Werte erreicht haben. Bei der Individualisierbarkeit gab es Abzüge, weil die Bedienelemente statisch aufgebaut sind und sich nicht individuell anpassen lassen. In einer Ausbaustufe kann das jedoch recht einfach angepasst werden.

Interview

Durch die anschließenden Interviews kam heraus, dass die Testpersonen ihre Aufgaben zufriedenstellend lösen konnten. Die Steuerung wurde anfangs von keinem perfekt durchgeführt, jedoch zeigte sich nach kürzester Zeit eine gute Verbesserung. Demnach ergibt sich für den Prototyp eine hohe Lernkurve, die eine effektive und effiziente Steuerung der Medieninhalte schnell möglich macht. Diese Schlussfolgerung lässt sich auch durch nachfolgende Tabelle verifizieren. In dieser ist zu sehen, wie viele Mausclicks auf der Programmoberfläche die Testpersonen durchschnittlich zur Lösung der jeweiligen Aufgaben gebraucht haben und wie viel Zeit dafür benötigt wurde. Aus der Anzahl der Mausclicks lässt sich ableiten, wie schnell und gut die gewünschten Funktionen erreichbar sind, demnach wie effektiv die Aufgaben ausgeführt werden konnten. Die Dauer im Verhältnis zu den benötigten Mausclicks steht für die Effizienz der Bearbeitung. Man sollte beachten, dass diese Werte relativ hoch sind, weil das Programm für die Testpersonen neu war. Ist der Benutzer eingearbeitet, so verringern sich diese Werte noch wesentlich. Das ist auch in den letzten beiden Spalten zu sehen, in denen Werte von geübten Benutzern gemittelt wurden.

	Klicks \emptyset	t \emptyset	Klicks 2 \emptyset	t 2 \emptyset
Einrichten der Videowand	5	34s	4	19s
Sponsor-Bild einrichten	8	41s	6	19s
Hinzufügen von 5 Medien	8	43s	5	23s
Manueller Übergang mit bestimmten Effekt	6	15s	5	7s
Ändern der Anzeigedauer eines Mediums	6	22s	5	8s
Hinzufügen eines Videostreams	6	52s	5	28s

Tabelle 8.1: Anzahl der benötigten Mausclicks und Dauer für die jeweiligen Aufgaben und einen 2. Durchlauf (gerundet).

Nach den jeweiligen Interviews stellten sich einige Punkte des Prototypen heraus, die in weiteren Entwicklungsstufen noch ausgebaut werden könnten. Nachfolgend werden die Wichtigsten von ihnen genannt:

1. Um ein Medienobjekt direkt oder als nächstes auf der Videowand anzeigen zu lassen, wurde eine Drag and Drop-Funktion vorgeschlagen. Hier könnte man das Objekt aus der Liste auf das jeweilige Vorschauenfenster ziehen und somit einen direkten Übergang erzeugen.

2. Um die Bedienung noch einfacher zu gestalten, sollte eine generelle Drag and Drop-Funktion eingeführt werden, durch die Dateien vom Betriebssystem in die Medienobjekt-Liste gezogen werden können.
3. Um ein Objekt in der Liste schneller identifizieren zu können wäre es sinnvoll, eine Direkt-Vorschau zu bekommen. Diese könnte in einem Tooltip angezeigt werden. Eine andere Möglichkeit wäre die Einführung einer neuen Medienauflistung. Diese könnte verschiedene Anzeigemodi wie Liste, Miniaturansichten oder Coverflow anbieten.
4. Die Ressourcen-Verwaltung von Flash beim Laden von vielen Bildern ist etwas ineffizient. So werden alle Bilder in den Arbeitsspeicher geladen, wenn sie dem Prototypen hinzugefügt werden. Bei einer großen Menge von Bildern führt dies teilweise zum Abstürzen des Programms.

8.4 Zusammenfassung

Die anfangs gestellten Probleme wurden durch die Methodik zufriedenstellend gelöst. Auch in den Interviews kam heraus, dass die Testpersonen alle Aufgaben mit dem Prototypen zu ihrer Zufriedenheit lösen konnten. Die Aufgaben wurden im Durchschnitt auch recht schnell und mit wenigen Fehlern gelöst. Wie auch bei anderer Software ist eine Einarbeitungsphase notwendig, die jedoch recht kurz ausfällt. Nach dieser kann noch effizienter gearbeitet werden.

Das Bedienkonzept wurde von den meisten Testpersonen als angenehm und als passend empfunden. Die Personen, die Videomischer bereits kannten, nutzten die Fadebar recht häufig. Andere Personen kamen auch ohne sie gut zurecht.

Kapitel 9

Zusammenfassung und Ausblick

Zielsetzung dieser Arbeit war es, die Anforderungen von Sportveranstaltungen im Hinblick auf Videowände zu analysieren und aus den daraus gewonnenen Erkenntnissen Konzepte zu entwickeln, die prototypisch in einer Software umgesetzt werden sollen. Insbesondere sollten Effektivität, Effizienz und Anwenderzufriedenheit beachtet werden, welche eine wichtige Grundlage für die Mensch-Maschine-Schnittstelle darstellen.

Schon durch die Einleitung wurde ersichtlich, dass ein Schwerpunkt der zu entwickelnden Konzepte bei dem Echtzeitaspekt liegt. Wenn Multimediainhalte auf einer Videowand präsentiert werden sollen, ist eine Bedienung derselben in Echtzeit notwendig, um eine gute Präsentation ermöglichen zu können. Auch die Rahmenbedingung der Sportveranstaltungen stellte gezielte Anforderungen in Bezug auf Echtzeit.

Hierzu wurde über den aktuellen Stand der Forschung aufgezeigt, dass es bisher keine wissenschaftlichen Untersuchungen zu diesem speziellen Bereich der Echtzeitbedienung gibt. Auch die bisher existierende Software bietet nur eine minimale Grundlage für die zu entwickelnden Konzepte.

Daraufhin wurden Grundlagen für die zu entwickelnden Konzepte vermittelt. Ein wichtiger Punkt bei Sportveranstaltungen sind Live-Übertragungen, zu denen erläutert wurde, wie man die Kamerabilder live auf den Computer übertragen kann. Als weiterer wichtiger Grundlagenteil wurde der Videomischer betrachtet. Dieser stellt das genaue Gegenstück zu einer Software für Videowände dar, nämlich eine hardwareseitige Alternative. Hier wurden allgemeine Bedienelemente und Konzepte näher erläutert.

Für den Prototyp wurde eine entsprechende Programmierumgebung benötigt, die den gewünschten Anforderungen entspricht. Hierzu zählten die Anforderungen nach Echtzeit und nach multimedialen Fähigkeiten. Für den Prototypen wurde deshalb Flash in Kombination mit Flex ausgewählt.

Im Hauptteil der Arbeit wurden dann benutzerseitige Konzepte für die Bedienung eines Echtzeitsystems entwickelt. Hierzu wurden die Anforderungen an das System unter den Aspekten Echtzeit, Sport und Multimedia analysiert und dann in Konzepten umgesetzt. Durch diese Untersuchungen hat sich herausgestellt, dass der Videomischer als Alternative für eine Software gute Bedienkonzepte anbietet, die auch eine gute Grundlage für eine Software darstellen. Deshalb wurden die Bedienkonzepte des Videomischers betrachtet und

die nützlichsten auf Softwareebene übertragen. Hierzu zählten der Bus, die Fadebar und das Vorschauenfenster. Dazu wurden technische Konzepte und eine Benutzeroberfläche entwickelt, welche die Grundlage für eine Echtzeitbedienung darstellt.

Parallel dazu wurden die entwickelten Konzepte in einem Software-Prototypen umgesetzt. Aus den wichtigsten Bedienelementen heraus wurden drei große Bereiche auf der Benutzeroberfläche umgesetzt: Die Medienliste, der Übergangsbereich mit Schieberegler und der Vorschaubereich. Unter Beachtung des sportlichen Aspekts wurde für Rad- und Laufveranstaltungen ein Modul entwickelt, das Top Ten-Ergebnisse von einem Webservice empfangen und auf der Videowand anzeigen lassen kann.

Die Ergebnisse haben gezeigt, dass die Zielvorgabe erreicht worden ist. Nach dem Vergleich mit bisheriger Software und der Benutzerbefragung der Evaluation, erfüllt der Prototyp die Zielvorgabe nach Effektivität, Effizienz und Anwenderzufriedenheit. In weiteren Untersuchungen könnte die Bedienung jedoch noch deutlich verbessert werden. Manche Funktionen wie Drag and Drop oder eine Direktvorschau der einzelnen Medienobjekte kann die Bedienung an verschiedenen Punkten noch deutlich vereinfachen und intuitiver gestalten.

Die Nutzung von Videowänden auf Sportveranstaltung wird auch künftig weiter zunehmen. Durch die fallenden Preise für Videowände werden auch immer mehr Großanzeigen auf kleineren Veranstaltungen eingesetzt. Um die softwareseitige Lücke zu schließen, ist die Untersuchung dieser Arbeit demnach für diese Veranstaltungen sehr interessant. Sowohl für die Veranstalter, als auch für die Anbieter um eine kostengünstige Alternative zu einem großem Hardwaresystem anbieten zu können.

Für die Weiterentwicklung des Systems sollten die Ergebnisse der Evaluation umgesetzt werden. Demnach sollte die Individualisierbarkeit dahingehend verbessert werden, dass sich die einzelnen Bedienelemente beliebig anordnen lassen. Auch Funktionen zur Unterstützung der Bedienung können ausgebaut werden. Vor allen Dingen ist eine Entwicklung von weiteren Modulen für mehrere Sportarten notwendig, um die Software für alle Sportarten attraktiv zu machen.

Anhang A

Inhalte der CD

Auf der beiliegenden CD sind die Bachelorarbeit und der Prototyp mit zugehöriger Anleitung und als Quellcode enthalten. Die CD ist folgendermaßen aufgebaut:

Adobe AIR

Dieser Ordner enthält die AIR-Laufzeitumgebung. Diese muss vor Installation der Anwendung auf dem System installiert sein.

Anleitung

Hier befindet sich eine kleine Anleitung für die Anwendung. Diese ist auch über die Hilfe-Funktion aus der Anwendung aufrufbar.

Bachelorarbeit

In diesem Ordner befindet sich das PDF-Dokument der Bachelorarbeit.

Dokumentation

Hier befindet sich die Dokumentation der einzelnen Klassen.

Evaluation

Die Unterlagen zur Evaluation befinden sich in diesem Ordner.

LaTeX

Hier befindet sich der LaTeX-Quellcode der Bachelorarbeit.

Prototyp

Hier befindet sich die Installationsdatei der Anwendung. Zum installieren und ausführen wird die AIR-Laufzeitumgebung aus dem Ordner Adobe AIR benötigt.

Quellcode

In diesem Verzeichnis ist der Quellcode der Anwendung enthalten. Dieser Ordner kann als Flex Projekt in den Flash Builder 4 importiert werden.

Anhang B

Actionscript 3 und MXML

Hier wird gezeigt, wie grafische Komponenten über AS3 und MXML erzeugt werden können. Die Verwendung von MXML stellt sich hier als sehr vorteilhaft heraus.

Definition eines Labels

```
<!-- Label-Komponente mit MXML -->
<s:Label x="100" y="200" label="Mein Text" id="label1"/>
```

```
// Label-Komponente mit AS3
import mx.controls.Label;

protected function init():void
{
    var label1:Label = new Label();
    label1.text = "Mein Text";
    label1.x = 100;
    label1.y = 200;
    this.addElement(label1);
}
```

Über MXML kann eine Label-Komponente durch XML-Tags beschrieben werden. Über AS3 ist das etwas umständlich und sollte nur bei dynamischen Elementen verwendet werden.

Kombination AS3 mit MXML

Eleganteste Lösung — Kombination von AS3 und MXML, um eine Meldung auszugeben: Bedienelemente sollten in MXML realisiert und die Programmlogik in AS3 entwickelt werden.

```
<!-- Button-Komponente mit MXML -->
<s:Button x="100" y="200" label="Meldung" id="btnMeldung" click="btn_click(event)"/>

// Programmlogik, die beim Drücken vom Button ausgeführt wird
protected function btn_click(event:MouseEvent):void
{
    Alert.show("Das ist eine Infomeldung","Info");
}
```


Glossar

AIR	Ist eine Laufzeitumgebung von Adobe, um Rich Internet Applications auf dem Desktop ausführen zu können.
AS3	Steht für Actionscript 3 und ist eine objektorientierte Programmiersprache von Adobe, die in Flash-Software verwendet werden kann.
Benchmark	Ein Benchmark beschreibt Verfahren, über die verschiedene Systeme miteinander verglichen werden können.
Client	Ist ein Computerprogramm, das Verbindung zu einem anderen Programm, dem Server aufnimmt, um verschiedene Daten auszutauschen.
Fadebar	Die Fadebar bezeichnet den Steuerregler an einem Videomischer, über den ein Übergang zum nächsten Eingabesignal erzeugt werden kann.
Form follows Function	Ist ein Leitsatz, nach dem sich die Gestaltung von Dingen aus ihrer Funktion und dem Nutzungszweck ergeben soll.
Joy of Use	Steht für den Nutzerspaß, den ein Benutzer beim Verwenden einer Software haben kann.
Laufzeitumgebung	Eine Laufzeitumgebung macht Anwendungsprogramme lauffähig, die nicht direkt mit dem Betriebssystem kommunizieren können. Hierzu vermittelt es die Informationen zwischen dem Programm und dem Betriebssystem.
LED	Kurzbezeichnung für light-emmiting diode, deutsch Leuchtdiode.
Linsn	Die Firma Linsn ist eine der größten Unternehmen im Bereich der LED-Wand Steuerung.
Multimedia	Der Begriff Multimedia bezeichnet Inhalte, die aus mehreren digitalen Medien bestehen wie Bild, Audio, Video, Grafik oder Text.
Modalität	Sinnesmodalitäten sind Empfindungskomplexe wie das Sehen und Hören.

MXML	Ist eine XML-Auszeichnungssprache für Flex-Inhalte. Mit MXML können ähnlich wie in HTML, visuelle Elemente beschrieben werden.
Pixel	Die kleinste Einheit zur Definition eines Bildpunkts ist der Pixel.
RIA	Rich Internet Applications sind Anwendungen im Internet, die Desktopanwendungen sehr ähneln und auch teilweise außerhalb des Browsers laufen können.
Rich-Text	Das Rich-Text Format wird zur Darstellung formatierter Texte verwendet.
SDI	SDI steht für Serial Digital Interface und ist eine digitale Schnittstelle, die zur Übertragung von Videodaten im Profibereich verwendet wird.
Tweens	Durch Tweening können Animationen in Flash erzeugt werden. Diese werden in der Kurzform Tweens genannt.
Video on Demand	Über Video on Demand, können Videofilme über einen Stream aus dem Internet angesehen werden.
Webservice	Ein Webservice ist ein Dienst im Internet, der verschiedenste Daten über eine XML-Schnittstelle, mit anderen Anwendungen austauschen kann.

Literaturverzeichnis

- [92498] 9241-11, DIN: *Anforderungen an die Gebrauchstauglichkeit DIN EN ISO 9241-11*. 1998
- [92406] 9241-110, DIN: *Grundsätze der Dialoggestaltung DIN EN ISO 9241-110*. 2006
- [ABCB02] ASSFALG, Jürgen ; BERTINI, Marco ; COLOMBO, Carlo ; BIMBO, Alberto D.: *Semantic Annotation of Sports Videos / University of Florence, Italy*. 2002. – Forschungsbericht
- [BLMS08] BAUMANN, Claudia E. ; LINDER, Philippe ; MEHR, Rebekka ; STOFER, Christian: *Wirtschaftliche Bedeutung der Sportveranstaltungen in der Schweiz / ITW Institut für Tourismuswirtschaft*. 2008. – Forschungsbericht
- [CZK01] CHANG, Shih-Fu ; ZHONG, Di ; KUMAR, Raj: *Real-Time Content-Based Adaptive Streaming of Sports Videos / Columbia University, NY*. 2001. – Forschungsbericht
- [Dah05] DAHM, Markus: *Grundlagen der Mensch-Computer-Interaktion*. Pearson Studium, 2005. – ISBN 978-3-8273-7175-1
- [Hat00] HATSCHER, Michael: *Joy of use - Determinanten der Freude bei der Software-Nutzung / Universität Osnabrück*. 2000. – Forschungsbericht
- [HB01] HARDENBERG, Christian von ; BERARD, Francois: *Bare-Hand Human-Computer Interaction / Technische Universität Berlin*. 2001. – Forschungsbericht
- [HW09] HITZENBERGER, Ludwig ; WOLFF, Christian: *Mensch-Maschine-Interaktion*. 2009. – Hochschultag Niederbayern-Oberpfalz – Info-Workshop
- [Iss02] ISSING, Ludwig J.: *Information und Lernen mit Multimedia und Internet*. Beltz-PVU, 2002. – ISBN 978-3-6212-7449-4
- [Nic90] NICOLAOU, Cosmos: *An Architecture for Real-Time Multimedia Communication Systems*. In: *IEEE journal on selected areas in communications* (1990)
- [Par11] PARTECKE, Ingo: *Videowalls und Anzeigetafeln Anbieterverzeichnis*. Brühl, 03 2011

- [PHPP11a] PARTECKE, Ingo ; HEIMES, Sven ; PAUER, Christopher ; PRÜMPER, Jan: LED: kleiner Baustein für große Bilder. In: *Videowalls und Anzeigetafeln 2011* (2011)
- [PHPP11b] PARTECKE, Ingo ; HEIMES, Sven ; PAUER, Christopher ; PRÜMPER, Jan: Wie die Bilder laufen lernen. In: *Videowalls und Anzeigetafeln 2011* (2011)
- [PJ00] PERSA, Stelian ; JONKER, Pieter: Human-Computer Interaction Using Real Time 3D Hand Tracking. In: *PROC. 21ST SYMPOSIUM ON INFORMATION THEORY IN THE BENELUX*, 2000, S. 25–26
- [Ree04] REEPS, Inga E.: *Joy-of-Use - eine neue Qualität für interaktive Produkte*, Universität Konstanz, Diplomarbeit, 2004
- [RF10] RICHTER, Michael ; FLÜCKIGER, Markus: *Usability Engineering kompakt*. Spektrum Akademischer Verlag, 2010. – ISBN 978-3-8274-2328-3
- [SCS⁺94] SEO, Bemjoo ; COVELL, Michele ; SPASOJEVIC, Mijana ; ROY, Sumit ; ZIMMERMANN, Roger ; KONTOTHANASSIS, Leonidas ; BHATTI, Nina: Evaluating Server Capacity for Streaming Media Services / University of Southern California, Los Angeles and Hewlett-Packard Laboratories and National University of Singapore. 1994. – Forschungsbericht
- [SP05] SHNEIDERMAN, Ben ; PLAISANT, Catherine: *Designing the user interface*. Pearson Education, 2005. – ISBN 0-321-19786-0
- [VLAV94] VINA, A. ; LERIDA, J. L. ; A.MOLANO ; VAL, D. del: Real-Time Multimedia Systems / Universidad Autonoma de Madrid. 1994. – Forschungsbericht
- [Wil05] WILTS, Henning: Sportevents und Nachhaltigkeit - Sport-Events als Massenmarkt / Wuppertal Institut für Klima, Umwelt, Energie GmbH. 2005. – Forschungsbericht
- [Zub02] ZUBERBÜHLER, Hans-Jörg: Wahrnehmung von Verzögerung in netzvermittelter Kommunikation / TH, Eidgenössische Technische Hochschule Zürich, Institute of Hygiene and Applied Physiology. 2002. – Forschungsbericht