

Fachhochschule Köln
University of Applied Science Cologne
Campus Gummersbach
Fakultät für Informatik und Ingenieurwissenschaften

Studiengang Medieninformatik

**„Optimierung des Metadatenmanagements eines
Fernsehsenders zur verbesserten Archiv-Recherche“**

BACHELORARBEIT

zur Erlangung des Grades
Bachelor of Science
in der Fachrichtung Medieninformatik

von: Alexander Stroick
Matrikelnummer: 110 455 83
Erstprüfer: Prof. Dr. rer. nat. Wolfgang Konen
Zweitprüfer: Dirk Sarembe, Cologne Broadcasting Center GmbH
Projektleiter TRIP

vorgelegt: 10. August 2007

Inhaltsverzeichnis

Abkürzungsverzeichnis	IV
Abbildungsverzeichnis	V
1 Einleitung	1
1.1 Motivation	1
1.2 Ziele und Ergebnisse der Arbeit	2
1.3 Der Privatsender RTL Television GmbH	3
1.4 Der Dienstleister Cologne Broadcasting Center GmbH	4
1.5 Die Marketingeinheit RTL Creation GmbH	4
2 Grundlagen	6
2.1 Was sind Metadaten?	6
2.2 Metadaten im Archivsystem eines Fernsehsenders	7
2.3 Wie kann man Metadaten unterteilen?	7
2.4 Wie ist eine Sendung strukturiert?	8
2.5 Das Archivsystem <i>TRIP</i>	9
2.5.1 <i>TRIP</i> Client	11
2.5.2 <i>TRIP</i> Highway	13
3 Problemanalyse und Soll-Konzeption	15
3.1 Recherche nach digitalem Videomaterial	15
3.1.1 Suchmaskeneingabe	15
3.1.2 Ausgabeformate	16
3.1.3 Weiterverarbeitung	22
3.2 Probleme und Engpässe	23
3.2.1 Aufwand	23
3.2.2 Inflexibilität	25
3.2.3 Kosten	25
3.3 Rahmenbedingungen	25
3.4 Lösungsansätze	26
3.4.1 Erweiterung der Recherchemöglichkeit	26
3.4.2 Neuentwicklung des Servermoduls	29
4 Umsetzung	32
4.1 Archivübergreifende Recherche	32
4.1.1 Optimierung der Webanwendung	32
4.1.2 Schwierigkeiten bei der Implementierung	41
4.1.3 Test und Qualitätssicherung	42
4.2 Bildvergleich der Vorschaubilder	42
4.2.1 Bildvergleich-Verfahren	43
4.2.2 Histogramm-Analyse	45
4.2.3 Abschließende Bewertung	57
4.2.4 Integration in die Webanwendung	59
4.3 Portierung des Servermoduls <i>TRIP</i> Highway	61
4.3.1 Funktionen der Eigenentwicklung	62

4.3.2 Aufgetretene Probleme und Abgrenzung	70
4.3.3 Konzept für ein erweitertes DB-System	71
5 Zusammenfassung und Ausblick	74
6 Anhang	75
6.1 Begriffsbestimmungen und Definitionen	75
6.1.1 Ampelsystem	75
6.1.2 Broadcast	76
6.1.3 VPMS PreviewClient	76
6.1.4 Spitzen und Löcher	77
6.2 Entwicklungswerkzeuge.....	77
6.2.1 Microsoft Visual Studio 2005.....	77
6.2.2 Microsoft Visual Source Safe.....	78
6.2.3 Primalscript.....	78
6.2.4 ImageJ.....	79
6.3 Eingesetzte formale Sprachen	79
6.3.1 C++	79
6.3.2 CCL	80
6.3.3 HTML	80
6.3.4 JavaScript.....	81
6.3.5 Java	81
Literaturverzeichnis.....	82
Danke!.....	85
Eidesstattliche Erklärung	86

Abkürzungsverzeichnis

Fachbegriffe werden bei der ersten Verwendung ausgeschreiben, anschließend auch in abgekürzter Form, und zwar kursiv gesetzt, verwendet.

Bsp.: Cologne Broadcasting Center = *CBC*

ACH	Adaptive Color Histogram
API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
CBC	Cologne Broadcasting Center
CCL	Common Command Language
CGI	Common Gateway Interface
CSS	Cascading Style Sheets
DOM	Document Object Model
EDV	Elektronische Datenverarbeitung
HP	Hewlett-Packard
HTML	Hypertext Markup Language
ID	Identität
IEC	International Electrotechnical Commission
INES	Integrated News Editing System
ISO	International Standard Organisation
JPEG	Joint Photographic Experts Group
MOS	Media Object Server
OF	OutputFormat
ODBC	Open Database Connectivity
PB	Paintbox
PHP	Hypertext Preprocessor
S4M	Solutions for Media
TC	Timecode
TCA	Timecode Anfang
TCE	Timecode Ende
THW	<i>TRIP</i> Highway
TRIP	Text Retrieval and Information Program
URL	Uniform Resource Locator
VPMS	Video Production Management System
XML	Extensible Markup Language

Abbildungsverzeichnis

Falls Abbildungen nicht selber erstellt wurden, wird mit einer Fußnote am Ende der Bildunterschrift auf die entsprechende Quelle verwiesen.

Abbildung 2-1: Sendungsstruktur eines Beitrags.....	9
Abbildung 2-2: TRIP Architektur	10
Abbildung 2-3: Eingabemaske des Newsarchivs.....	12
Abbildung 2-4: Systemarchitektur des Softwaresystems TRIPHighway	14
Abbildung 3-1: Screenshot der TRIPHighway Suchmaske.....	15
Abbildung 3-2: Titelformat TRIPHighway.....	16
Abbildung 3-3: Übersicht Sendungskürzel	17
Abbildung 3-4: Langformat TRIPHighway	18
Abbildung 3-5: Sequenzformat TRIPHighway.....	20
Abbildung 3-6: Videoformat TRIPHighway	21
Abbildung 3-7: Textformat TRIPHighway.....	22
Abbildung 3-8: produzierte Elemente von RTL Creation.....	23
Abbildung 3-9: Vor- und Nachteile Lösungsansatz 1.....	26
Abbildung 3-10: Vor- und Nachteile Lösungsansatz 2.....	27
Abbildung 3-11: Vor- und Nachteile Lösungsansatz 3.....	28
Abbildung 3-12: Servermodul Vor- und Nachteile Lösungsansatz 1	29
Abbildung 3-13: Servermodul Vor- und Nachteile Lösungsansatz 2	30
Abbildung 3-14: Servermodul Vor- und Nachteile Lösungsansatz 3	30
Abbildung 4-1: Auswahl des Feldes im Ausgabeformat	32
Abbildung 4-2: Codeauszug Erzeugung des Links.....	33
Abbildung 4-3: Ausgabe des Langformats vor Formatierung	34
Abbildung 4-4: Ausgabe des Langformats nach Formatierung.....	35
Abbildung 4-5: Vergleich Sendungskürzel.....	36
Abbildung 4-6: Workflow "Suchanfrage generieren"	37
Abbildung 4-7: Frameunterteilung TRIPHighway	38
Abbildung 4-8: Ausgabe des Suchergebnisses des Creationarchivs.....	39
Abbildung 4-9: Suchmaske Creationarchiv	41
Abbildung 4-10: Testbilder Bildvergleich	44
Abbildung 4-11: Vergleich der Luminanz-Histogramme	46
Abbildung 4-12: Berechnung der Mittelwert-Differenzen	48
Abbildung 4-13: Formel - prozentuales Vergleichsmaß.....	48
Abbildung 4-14: Formel - Vergleichsmaß	49
Abbildung 4-15: Ergebnis des "Vergleich der Mittelwerte".....	50
Abbildung 4-16: Beispiel für korrekte Zuordnung beim Mittelwertvergleich.....	51
Abbildung 4-17: Berechnung der Absolutwert-Differenzen.....	52
Abbildung 4-18: Ergebnis des "Vergleich der Absolutwerte".....	53
Abbildung 4-19: Beispiel für korrekte Zuordnung beim Absolutwertvergleich.....	53
Abbildung 4-20: Berechnung der Projektion-Differenzen.....	54
Abbildung 4-21: Ergebnis des "Vergleich der 2D-Farbhistogramme"	55
Abbildung 4-22: Beispiel für korrekte Zuordnung beim 2D-Histogrammvergleich	56
Abbildung 4-23: Kombination der Differenzen.....	56
Abbildung 4-24: Ergebnis des kombinierten Vergleichs.....	57
Abbildung 4-25: Beispiel für nicht auffindbare Bildpaare	58

Abbildung 4-26: Datenflußdiagramm Bildvergleich	60
Abbildung 4-27: Portierung TRIPHighway	62
Abbildung 4-28: Beispiel URL TRIPHighway	63
Abbildung 4-29: Schlüsselwörter in der TRIPHighway URL	65
Abbildung 4-30: Schlüsselwörter in den THW Templates	68
Abbildung 4-31: Beispiel Template-Datei TRIPHighway	69
Abbildung 4-32: Beispielkonzept für Datenbankanbindung	71
Abbildung 4-33: Klasse Connection	72
Abbildung 6-1: Ampelsystem am Beispiel eines Beitrags über "Natascha Kampusch"	75
Abbildung 6-2: Vorerfassung eines Beitrags durch VPMS PreviewClient	76
Abbildung 6-3: Spitzen und Löcher im Histogramm	77

1 Einleitung

1.1 Motivation

Im Hause von RTL werden tagtäglich über 100 audiovisuelle Beiträge in dem Bereich Information (Nachrichten, Sport, Magazine) produziert. Jeder Beitrag wird durch eine Vielzahl von Metadaten¹ beschrieben. Insgesamt haben sich seit 1984 über 580.000 Beiträge und deren beschreibende Daten alleine in diesem Bereich angesammelt. Bei einer solchen Datenmenge ist eine strukturierte Verwaltung der Metadaten (sogenanntes Metadatenmanagement) zwingend erforderlich.

Das wesentliche Management teilt sich in zwei große Bereiche: Erfassung von audiovisuellen Medien und Recherche nach archivierten Beiträgen.

Diese Bachelorarbeit beschreibt im ersten Teil die Optimierung beim firmenübergreifenden Zugriff auf die Metadaten durch einen WebClient namens *TRIPHighway*². Die dadurch ausgelöste schnittstellenübergreifende Zusammenarbeit zwischen verschiedenen Archiven ist besonders wichtig, um Arbeitsabläufe innerhalb der Produktion zu vereinfachen und den Zeitaufwand bei der Informationssuche zu minimieren.

Im zweiten Teil wird die Portierung³ des lizenzpflichtigen *TRIPHighway*-Servermoduls zu einer Eigenentwicklung beschrieben. Dieses Vorgehen schafft RTL eine größere Herstellerunabhängigkeit und spart zusätzlich Lizenz- und Wartungskosten. Weiterhin können durch das neu entwickelte Servermodul gleichzeitig Anpassungen und Erweiterungen implementiert werden, um auf zukünftige Anforderungen möglichst flexibel zu reagieren.

¹ weitere Informationen im Kapitel 2.1 „Was sind Metadaten?“

² weitere Informationen im Kapitel 2.5.2 „*TRIPHighway*“

³ Ablösung oder Wechsel

1.2 Ziele und Ergebnisse der Arbeit

Das Ziel und die Herausforderung dieser Arbeit ist, schnittstellenübergreifendes Arbeiten zwischen der Firma RTL Television, der Firma RTL Creation und den beteiligten Archiven zu ermöglichen.

In diesem Zusammenhang gilt es, das Metadatenmanagement in dem Bereich Information des Unternehmens RTL Television und der Tochterfirma RTL Creation zu untersuchen (siehe Kapitel 3), Schnittstellen aufzuzeigen und die daraus resultierenden Anforderungen zu implementieren (siehe Kapitel 4). Die Änderungen und Erweiterungen werden anhand praktischer Beispiele dargestellt und erläutert.

Die Arbeit zeigt, dass es bei RTL Television und RTL Creation trotz einer strukturierten Verwaltung der Metadaten in audiovisuellen Medien immer noch Schwachstellen gibt, die zu beseitigen sind.

Arbeitsabläufe bei der Erfassung, Dokumentation und Recherche von Archivmaterial sind stringent, zwischen den einzelnen Archiven aber unterschiedlich und voneinander getrennt. Diese Unterschiede wurden bei der Umsetzung berücksichtigt. Somit wurde eine schnittstellenübergreifende Suche zwischen dem Creation- und Newsarchiv mit der Webanwendung *TRIPHighway* ermöglicht ohne den Rechercheaufwand gravierend zu erhöhen.

Der praktische Teil der Bachelorarbeit beinhaltet neben der Realisierung der archivübergreifenden Recherche (siehe Kapitel 4.1), die Entwicklung eines Bildvergleichsverfahrens (siehe Kapitel 4.2). Durch den Vergleich der Vorschaubilder ist es zusätzlich möglich, nach relevanten Treffern bei der Präsentation der Suchergebnisse zu sortieren. Das Bildvergleich-Modul ist zu Testzwecken in die Anwendung integriert; bis Ende August 2007 wird eine endgültige Entscheidung über dessen Einsatz in *TRIPHighway* getroffen.

Weiterhin wurde das lizenzpflichtige Servermodul weitestgehend durch eine Eigenentwicklung abgelöst (siehe Kapitel 4.3). Zur vollständigen Portierung fehlt die Anpassung einzelner Serverkomponenten, die den Zugriff auf einzelne Felder innerhalb eines Datensatzes erlauben.

Zuletzt wurde ein Konzept für einen Datenbankzugriff entworfen, welches die Koppelung anderer Datenbanken an das Archivsystem zukünftig zulässt.

Insgesamt ist eine fundierte Basis für die schnittstellenübergreifende Zusammenarbeit zwischen dem News- und Creationarchiv geschaffen worden.

1.3 Der Privatsender RTL Television GmbH

Der 1984 gegründete Privatsender RTL ist eine Tochter der RTL Group Luxemburg und Deutschlands erfolgreichster Fernsehsender sowohl bei den Zuschauern als auch auf dem TV-Werbemarkt.

RTL hat sich in 23 Jahren Firmengeschichte zu einer großen Unternehmensgruppe entwickelt, die sich mit zahlreichen Engagements rund um das Kerngeschäft Fernsehen befasst. Zu dieser Unternehmensgruppe gehören:

- Beteiligungen an den Fernsehsendern n-tv, VOX, RTL II, Super RTL und RTL Shop
- Beteiligungen an den Regionalgesellschaften RTL Nord, RTL West, RTL Hessen
- Vermarktungsunternehmen IP Deutschland
- Agentur für Marketing, Promotion und Design RTL Creation
- Full-Service Agentur RTL Enterprises
- System- und Beratungshaus Solutions for Media (*S4M*)
- Beteiligungen an den Produktionsfirmen ActionConcept, Norddeich TV und Dienstleister Cologne Broadcasting Center (*CBC*)
- RTL Journalistenschule

Derzeit beschäftigt der Privatsender ca. 1000 feste Mitarbeiter und erreicht einen jährlichen Umsatz von ca. 1.844,00 Mio. Euro. Der Hauptsitz des Unternehmens befindet sich in Köln⁴. An diesem Standort sitzt ebenfalls der hauseigene IT-Dienstleister: RTL *EDV*⁵. Durch eine firmenpolitische Entscheidung wurde die *EDV* zum 1. März 2007 in die *CBC* GmbH ausgelagert.

⁴ vgl. [RTL 2007]

⁵ Elektronische Datenverarbeitung

1.4 Der Dienstleister Cologne Broadcasting Center GmbH

Das Produktions- und Broadcastingunternehmen *CBC* wurde 1994 in Köln gegründet und entstand aus der RTL Produktions GmbH. Die Gesellschaft wird von Thomas Harscheidt geleitet und ist heute einer der führenden Dienstleister innerhalb der Medienbranche.

Momentan beschäftigt die *CBC* 230 feste und 180 freie Mitarbeiter, die auf dem Hauptstandort Köln und den beiden anderen Standorten in München/Grünwald und Hannover verteilt sind.

Die Firma legt ihren Focus auf vielfältige und individuelle Serviceleistungen in der Fernsehbranche und unterteilt sich in sechs verschiedene Geschäftsbereiche: Sendebewicklung & Teleport, Systems & Technology (vorher *RTL EDV*), Produktion, Postproduktion, Media Services und das Facility Management.

Zusätzlich zu den Fullfillment⁶-Dienstleistungen betreut die *CBC* noch 50 lineare und nonlineare Schnittplätze (sogenannte Editsuiten), Grafikplätze und digitale Tonstudios, die an 365 Tagen im Jahr Postproduktion ermöglichen.⁷

Unternehmensbereich Systems & Technology – Produktionssysteme

Das seit dem 1. März in die *CBC* eingegliederte Ressort Systems & Technology (vorher *RTL EDV*) unterteilt sich in die Tätigkeitsfelder Kaufmännisches-, Channel-Management, Produktionssysteme, Benutzerservice, Netzservice und Datenbank/Technologie.

Die Mitarbeiter in dem Tätigkeitsfeld „Produktionssysteme“ befassen sich mit dem Konzipieren, Entwickeln und Administrieren von individuellen Softwarelösungen für den Produktionsbereich und der Betreuung und Beratung der internen Kunden in der Unternehmensgruppe RTL.

1.5 Die Marketingeinheit RTL Creation GmbH

Diese 100% Tochtergesellschaft des Senders RTL mit Sitz in Köln ist für die Gestaltung des Corporate Designs und die Entwicklung aller Werbemaßnahmen von RTL

⁶ „alle Leistungen aus einer Hand“

⁷ vgl. [CBC 2007]

verantwortlich. Hierzu zählen die On- und Off-Air-Aktivitäten der Bereiche Marketing, Promotion und Design.

Das Tätigkeitsumfeld der Firma reicht von klassischer Corporate Identity Arbeit, über die Konzeption und Mediaplanung, bis zur Produktion.

Zu den Kunden von RTL Creation zählen neben RTL ebenso die RTL Group, IP Deutschland GmbH, RTL Shop, n-tv, RTL Interactive, RTL Enterprises, etc.⁸

Das Unternehmen ist vollkommen eigenständig und verwaltet daher alle von ihnen produzierten Materialien getrennt von RTL in einem eigenen Creationarchiv.

⁸ vgl. [CREATION 2007]

2 Grundlagen

Im Anschluss an die kurze Einführung zur Problemstellung und den Zielen der Arbeit werden in diesem Abschnitt alle grundlegenden Begriffe rund um das Thema Metadatenmanagement eingeführt. Ziel des Kapitels ist es, eine fundierte Basis für die Problemanalyse zu schaffen.

2.1 Was sind Metadaten?

Eine der am häufigsten zitierte Begriffsbestimmung für Metadaten lautet: „Strukturierte Daten, die andere Daten definieren oder beschreiben, nennt man Metadaten“⁹.

Setzt man sich jedoch intensiver mit dem Thema auseinander, stellt man fest, dass eine Definition von Metadaten immer kontextabhängig ist. Anhand der folgenden Definitionen lässt sich erkennen, wie unterschiedlich Metadaten in verschiedenen Bereichen (z.B. Broadcast, Multimedia, Fernsehen, Software Engineering) beschrieben und erklärt werden.

Metadaten können den Einblick in syntaktisch oder semantisch komplexe Rohdaten vereinfachen, indem sie die Kernaussagen in Form von einfachen Beschreibungen festhalten. In den Broadcast Medien beinhalten Metadaten sowohl die Beschreibung der Dienste als auch die Beschreibung des eigentlichen multimedialen Inhalts.¹⁰

Im Bereich des Fernsehens werden die Daten als Metadaten bezeichnet, die für die effiziente Planung zur Erstellung und weitgehend automatisierten Bearbeitung der Nutzdaten (Video, Audio und Zusatzdaten), deren Verwaltung und Speicherung sowie deren Austausch erforderlich sind.¹¹

Im Umfeld des Software Engineering werden Metadaten nicht nur zur Dokumentation, sondern vor allem auch zur Automatisierung von Entwicklungsaufgaben eingesetzt.¹²

In dieser Arbeit sind mit Metadaten diejenigen Daten gemeint, welche die IT-basierte Verwaltung eines audiovisuellen Videos im Fernsbereich ermöglichen.

⁹ vgl. [ISO 2003]

¹⁰ vgl. [LuNiKa 2004], S. 3

¹¹ vgl. [Ebner 2005]

¹² vgl. [StaVaVe 1999]

2.2 Metadaten im Archivsystem eines Fernsehsenders

Die beschreibenden, strukturierten Metadaten sind in einem Archivsystem eines Fernsehsenders unentbehrlich. Ein Mediendokumentar beschreibt die einzelnen audiovisuellen Beiträge und legt sie im Archiv ab. Ein Rechercheur (z.B. ein Redakteur) hingegen muss mit Hilfe der dokumentierten Metadaten diese archivierten Beiträge meist unter Zeitdruck wiederfinden, um auf aktuelle Ereignisse zu reagieren.

Neben diesem naheliegenden Vorteil der besseren Wiederauffindung von Materialien, erschaffen Metadaten in gemeinsamen Informationsräumen (hier das Archivsystem von RTL) einen zusätzlichen indirekten Kommunikationskanal (Recherche mit Hilfe der Webanwendung). Sowohl die Dokumentationsarbeit als auch die Recherchearbeit kann somit voneinander zeitlich unabhängig (asynchron) stattfinden¹³ und erleichtert abteilungsübergreifendes Arbeiten.

2.3 Wie kann man Metadaten unterteilen?

Im Unternehmen RTL unterteilt man technische und inhaltsbezogene (semantische) Metadaten im Umfeld der Software *TRIP*¹⁴.

Die technischen Informationen beziehen sich auf konkrete und identifizierende Angaben zu dem Bewegtbild. In Kategorien¹⁵ eingeteilt, ergeben sich:

- Studio, Standort, Trägerformat (Allgemein)
- *TRIP*-Archivnummer, Material-*ID* (Identifizierung)
- Timecode-Anfang¹⁶, Timecode-Ende¹⁷, Dauer der Sendung (Timecode)
- Ton (Ton)
- Sendedatum, Erfassungsdatum, Bearbeitungsdatum (Datum)
- Key Frame (Video)

Beschreibende und funktionale Informationen zu dem Bewegtbild liefern dagegen die semantischen Metadaten:

¹³ vgl. [Steffens 2004], S.13

¹⁴ Text Retrieval and Information Program – weitere Informationen im Kapitel 2.5 „Das Archivsystem *TRIP*“

¹⁵ Kategorien entnommen aus [HeKuSaSi 2006], S.19-20

¹⁶ Abkürzung *TCA*

¹⁷ Abkürzung *TCE*

- Sendungsformat, Sendetitel, Beschreibung, Zusammenfassung, Deskriptor, Moderation, Dokumentar (Allgemein)
- Produktionsort, Redakteur, Autor, Kamera, Regie, Schriftgenerator (Produktion)
- Titel, Herkunft, Deskriptor, Bemerkungen, Agenturtext, Sparte (Beitragebene)
- Beschreibung, Land, Stadt, Herkunft, Topic, Sequenzrechte, Sequenzsperre, O-Ton (Sequenzebene)

2.4 Wie ist eine Sendung strukturiert?

Damit die Metadaten das audiovisuelle Medium eindeutig beschreiben, muss auch eine Sendungsstruktur existieren. Die Metadaten sind auch hier kontextbezogen und stark abhängig vom Aufbau der Sendung.

Eine Sendung teilt sich in mehrere Beiträge (inklusive Anmoderation) auf. Ein Beitrag besteht dabei aus mindestens einer Sequenz (eine Folge von Einheiten, die inhaltlich zueinander passen). Die Sequenz teilt sich wiederum in eine beliebig große Anzahl von Kameraeinstellungen (Shots) auf. Es muss auch hier mindestens ein Shot existieren. Ein Shot untergliedert sich in mehrere Frames (Einzelbilder), wobei mindestens ein Frame vorhanden sein muss (siehe Abbildung 2-1).

Mit Hilfe dieser hierarchischen Struktur kann eine Sendung in einzelne übersichtliche Teile zerlegt werden, so dass sie durch Metadaten beschrieben werden kann.

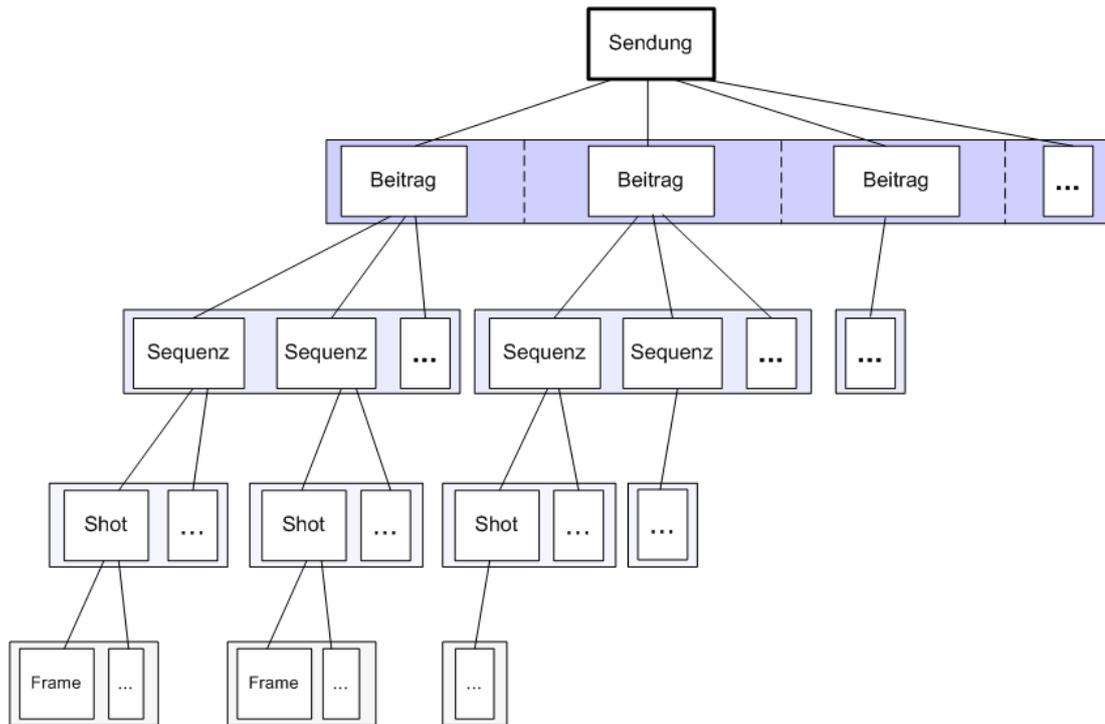


Abbildung 2-1: Sendungsstruktur eines Beitrags¹⁸

2.5 Das Archivsystem *TRIP*

Seit Mai 1996 werden bei RTL alle Metadaten, mit Hilfe der von der Firma Tieto-Enator (vorher TSI/PSI) entwickelten Volltextdatenbank namens *TRIP*, strukturiert archiviert.

Innerhalb des Unternehmens wird das Archivsystem aus Sicht der Benutzer in drei Bereiche unterteilt: Programmarchiv, Creationarchiv und Newsarchiv. Das Newsarchiv-System untergliedert sich zusätzlich noch in ein zentrales Archiv bei RTL in Köln und in die einzelnen Archive der Außenstudios: RTL II, RTL West, RTL Nord, n-tv, Berlin, München und Frankfurt. Das Programm- und das Creationarchiv sind nicht weiter untergliedert.

Diese Arbeit bezieht sich auf die Systeme des News- und Creationarchivs, da hier eine weitgehend identische DB-Struktur vorliegt. Die Benutzer des Programmarchivs und das Programmarchiv-System sind von der Umsetzung nicht betroffen.

Die Archivdaten werden durch die Software *TRIP* in einer Volltextdatenbank abgespeichert.

¹⁸ Erweiterung und Anlehnung an Bild 5 und 10 aus [Ebner 2005]

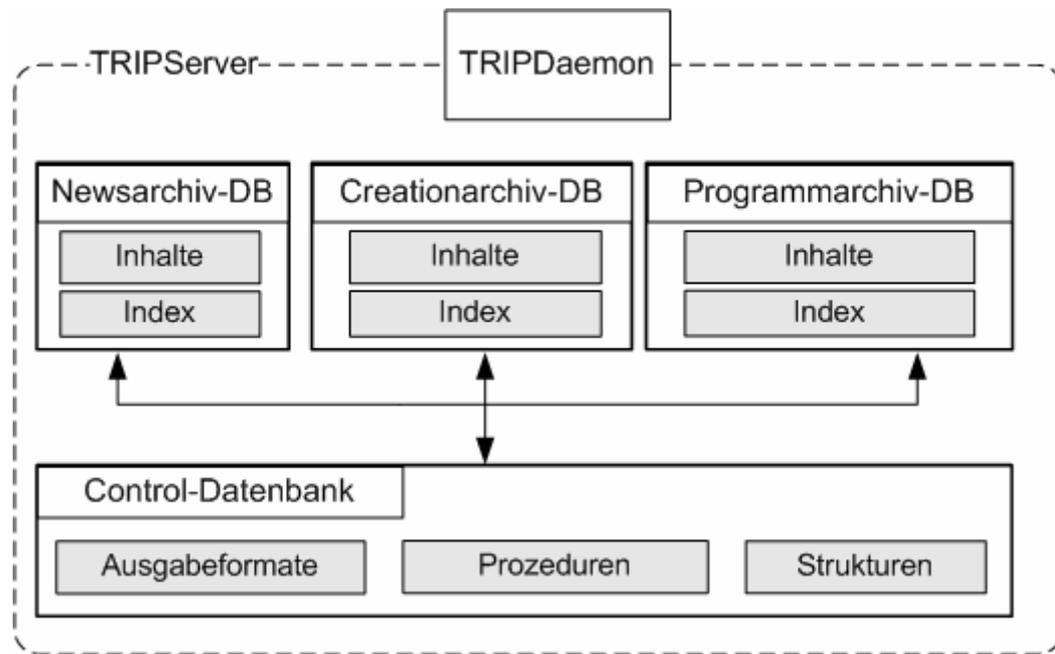


Abbildung 2-2: TRIP Architektur

Eine Archiv-Datenbank besteht aus den Inhalten, die ein Mediendokumentar einfügt und Informationen über die Indexierung der einzelnen Wörter (siehe Abbildung 2-2).

Zusätzlich gibt es eine Hauptdatenbank namens „Control-Datenbank“, die von TietoEnator geliefert wird und die Kernkomponente von *TRIP* ausmacht. Die darin enthaltenen Inhalte definieren die Strukturen aller anderen von RTL selbst angelegten Datenbanken. Außerdem werden dort Datenbankprozeduren und Ausgabeformate für Suchergebnisse festgelegt.

Die Ausgabeformate für die Suchergebnisse legen fest, wie ein Datensatz in der Datenbank ausgegeben wird. Dabei wird eine benutzerdefinierte Datenzusammenfassung produziert, die nur die Informationen enthält, welche der Benutzeranforderung entsprechen. Ein Ausgabeformat definiert welcher Datensatz ausgegeben wird, wie er formatiert wird und in welche Abschnitte die Informationen unterteilt sind.¹⁹

Sobald nun neue Informationen in der Datenbank abgespeichert werden, werden sowohl die Inhalte als auch der Index einer Datenbank aktualisiert. Beim Anlegen eines neuen Informationsspeichers werden zunächst die Inhaltsstrukturen in der Control-

¹⁹ vgl. [Tieto 2000]

Datenbank definiert. Weitere Datenänderungen können später nachindexiert werden und sind somit sehr schnell recherchierbar.²⁰

Um Suchanfragen zu formulieren und somit gespeicherte Inhalte abzurufen, steht die durch den *ISO*-Standard 8777 definierte Kommandosprache Common Command Language (*CCL*²¹) zur Verfügung.²²

Die Volltextdatenbank *TRIP* wurde ausgewählt, da sie den hohen Anforderungen an die Zugriffs- und Übertragungszeit (sogenannte Performance) bei der Recherche gerecht wird.²³

2.5.1 *TRIP* Client

Die Anwendung dient der Erfassung und Recherche von Bildmaterial in der Volltextdatenbank *TRIP* und wird von den Mitarbeitern des Programm-, Creation- und Newsarchivs benutzt.

Die Dokumentation des gesendeten Materials wird mit Hilfe von verschiedenen Eingabemasken durchgeführt, die für die Bedürfnisse der einzelnen Archive angepasst sind. In der Abbildung 2-3 sieht man die Eingabemaske für die Erfassung eines Nachrichtenbeitrags im Newsarchiv:

²⁰ vgl. [HeKuSaSi 2006], S.5-6

²¹ weitere Informationen im Kapitel 6.3.2 „CCL“

²² vgl. [HeKuSaSi 2006], S. 5-6

²³ vgl. [Saremba 1999]

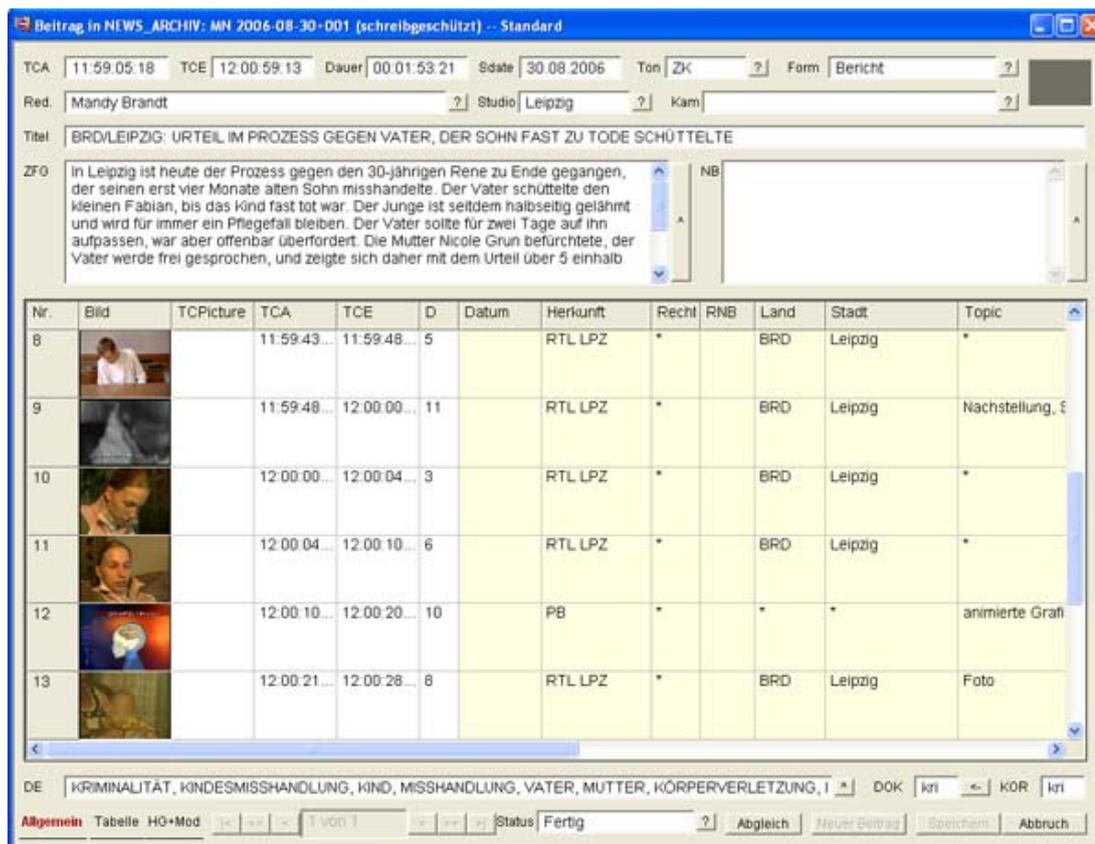


Abbildung 2-3: Eingabemaske des Newsarchivs

Die Recherche nach archivierten Beiträgen kann mit Hilfe von Suchmasken durchgeführt werden; dabei gibt es mindestens eine Suchmaske pro Datenbank. Die Suchergebnisse werden dann mittels *TRIP*-Ausgabeformaten formatiert und in einem eigenen Fenster dargestellt.

Die Anwendung ist sehr komplex, da sie neben der Schnittstelle zur Volltextdatenbank ebenfalls eine Schnittstelle zum Video Production Management System (*VPMS*) enthält. Das *VPMS* übernimmt die Vorerfassung der Nachrichtenbeiträge und übermittelt die Daten strukturiert an *TRIP*. Ebenso kann der Benutzer mit Hilfe des *TRIP* Clients einzelne Sequenzen auswählen und diese direkt in Vorschauqualität mit dem externen Player *VPMS PreviewClient*²⁴ anschauen.

Der *TRIP* Client steht nur den Dokumentaren zur Verfügung; seit dem Jahr 2001 können die Redakteure aber selber über die webbasierte Anwendung *TRIPHighway* recherchieren.

²⁴ weitere Informationen im Kapitel 6.1.3 „*VPMS PreviewClient*“

2.5.2 *TRIPHighway*

Die webbasierte Anwendung *TRIPHighway* ermöglicht den Redakteuren und allen sonstigen Interessenten eine Recherche über den Volltext des archivierten, digitalen Videomaterials.

Sie ist im Firmenintranet integriert und somit von jedem Arbeitsplatz mit Zugriff auf das Netzwerk erreichbar. Damit kann die Recherche unabhängig von den Arbeitszeiten und der Auslastung der Archivare durchgeführt werden.

Vor der Einführung der Software konnten die Redakteure allenfalls die Mediendokumentare telefonisch beauftragen, nach archiviertem Material zu recherchieren. Dadurch kam es ständig zu erheblichen Engpässen und hohen Wartezeiten, erst recht wenn die Nachrichtenlage, aufgrund von vielen aktuellen Ereignissen, angespannt war.

Die daraufhin entwickelte Webanwendung sollte die Archiv-Mitarbeiter entlasten und dabei helfen, dass auch nicht ausgebildete Mediendokumentare effizient nach Beiträgen recherchieren können.

Die Software enthält auf der einen Seite eine Suchmaske um die Suchbedingungen zu formulieren und einzugrenzen, auf der anderen Seite beinhaltet sie Ausgabeformate, die die Suchergebnisse je nach Anforderung entsprechend aufbereitet darstellen.

Aus welchen technischen Komponenten besteht TRIPHighway?

Diese Anwendung besteht clientseitig aus *HTML*²⁵-Seiten mit denen der Benutzer interagiert und mittels *HTML*-Formularen einzelne Suchkriterien zusammenstellt. Nachdem der Benutzer die Suchanfrage abschickt, wird mittels Javascript²⁶ ein sogenannter GET-Request in Form einer *URL*²⁷ an den Server gesendet (siehe Abbildung 2-4).

²⁵ Hypertext Markup Language; weitere Informationen im Kapitel 6.3.3 "HTML"

²⁶ weitere Informationen im Kapitel 6.3.4 „JavaScript“

²⁷ Uniform Resource Locator

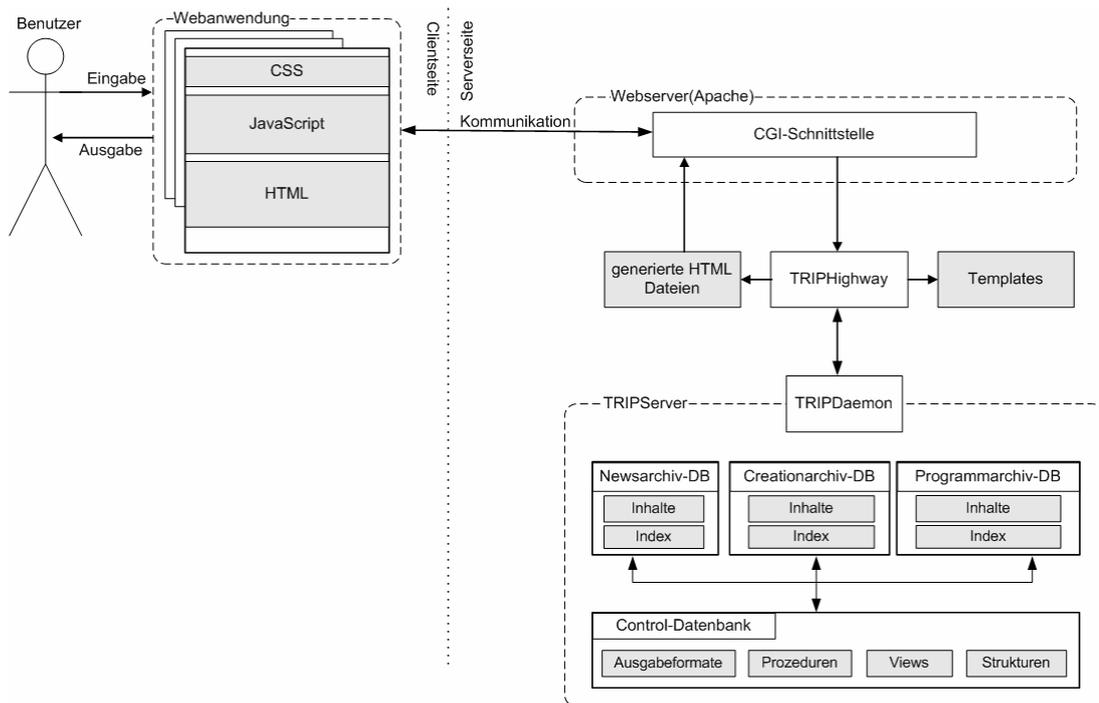


Abbildung 2-4: Systemarchitektur des Softwaresystems TRIPHighway

Der Webserver nimmt die in einer *URL* verpackte Suchanfrage entgegen und startet das von der Firma TietoEnator eingekaufte Programm *TRIPHighway* über die *CGI*²⁸. Der Anwendung wird dabei die generierte *URL* als Parameter übergeben.

Das Servermodul analysiert die übergebene *URL* und schickt die Suchanfragen in Form von *CCL* – Befehlen an die Volltextdatenbank *TRIP*.

Die Suche wird ausgeführt und das Suchergebnis wird in selbstdefinierten Ausgabeformaten an das *TRIPHighway* Programm zurückgegeben. Dieses füllt sogenannte *THW* Templates. Das sind *HTML*-Vorlagen, die *TRIP*-spezifische Syntax in Form von Schlüsselwörtern oder Schlüsselbedingungen (z.B. *TRIPIF*, *TRIPLOOP*) enthalten, die durch das Serverprogramm gelesen und verarbeitet werden können.

Die daraus generierten, vollständigen *HTML*-Seiten sendet der Webserver mittels einer Serverantwort (sogenannter *Response*) an den Browser des Anwenders.

²⁸ Common Gateway Interface

3 Problemanalyse und Soll-Konzeption

Im folgenden Kapitel wird das aktuelle Metadatenmanagement im Creation- und Newsarchiv vorgestellt und erläutert. Schwachstellen in der bestehenden Anwendung werden aufgezeigt und Lösungsansätze für eine Optimierung entwickelt.

3.1 Recherche nach digitalem Videomaterial

In diesem Abschnitt wird der momentane Recherche-Vorgang eines Redakteurs im Newsarchiv veranschaulicht. Hierbei wird ein praktisches Beispiel gewählt, dass in der Mittagsnachrichtensendung „Punkt 12“ ausgestrahlt wurde und einen großen Teil des Problemumfelds erklärt.

3.1.1 Suchmaskeneingabe

Abbildung 3-1: Screenshot der TRIPHighway Suchmaske

Mit Hilfe der Suchmaske (siehe Abbildung 3-1) sucht der Redakteur in diesem Beispiel nach einem Beitrag in der Datenbank des Newsarchivs [1], der in einem beliebigen Metadaten-Feld das Wort „Schüttel-Trauma“ [2] und in einem solchen Beitrag das Bild eines Babys [3] enthält.

Außerdem soll eine Preview (engl. Vorschau) [4] zu dem gesuchten Beitrag möglich sein und der Beitrag muss in Broadcast²⁹-Qualität [5] vorliegen.

Die Suche wird zusätzlich noch durch einen Zeitraum [6] eingeschränkt, in dem die Sendung ausgestrahlt wurde.

Ferner ist noch das Feld „Rechte - frei“ [7] aktiviert, womit nur Beiträge herausgesucht werden, die für den angegebenen Bereich „RTL Information“ [8] kostenlos zur Verfügung stehen.

Nach Verlassen eines Eingabefeldes wird die Trefferanzahl des jeweiligen gefüllten Feldes zurückgegeben. Bestätigt der Benutzer mit der „Enter“-Taste oder des „Such“-Buttons, dann wird die Gesamttrefferanzahl [9] berechnet. Anschließend kann er sich einzelne Suchergebnisse über die verschiedenen Ausgabeformate [10] im Detail anschauen.

3.1.2 Ausgabeformate

Das Titelformat. Dieses Format bietet relativ wenige Informationen zu einem einzelnen Suchergebnis. Es dient dazu, aus einer Vielzahl von Treffern (max. 499 Treffer) das richtige Ergebnis anhand dieser Metadaten auszuwählen:



Abbildung 3-2: Titelformat TRIPHighway

In diesem Beispiel gibt es nur drei Datensätze, die den weiter oben aufgeführten Suchbedingungen entsprochen haben. Zu jedem werden eine fortlaufende Nummer, die Datenbank, in der der Datensatz gefunden wurde und eine eindeutige Identität (*ID*) angegeben. Die *ID* setzt sich aus einem Kürzel der archivierten Sendung und dem Ausstrahlungsdatum in amerikanischer Notation zusammen. Hier folgt eine Liste der Sendungskürzel und dazu passenden Namen und Kategorien, die momentan für Beiträge aus dem Bereich Information verwendet werden:

²⁹ engl. Ausstrahlung - weitere Informationen im Kapitel 6.1.2 „Broadcast“

Sendungskürzel	Name der Sendung	Kategorie
EE	Extra	Magazin
EL	Life	Magazin
ET	Exklusiv	Magazin
EX	Explosiv	Magazin
FN	Punkt 6/7	Nachrichten
FN9	Punkt 9	Nachrichten
Form	Formel 1	Sport
Fuba	Fußball	Sport
MN	Punkt 12	Nachrichten
N	RTL Aktuell	Nachrichten
S-N	RTL Aktuell Sondersendung	Nachrichten
NJ	Nachtjournal	Nachrichten
S-NJ	Nachtjournal Sondersendung	Nachrichten
Ski	Ski-Sport	Sport
SN	Spätnachrichten	Nachrichten

Abbildung 3-3: Übersicht Sendungskürzel

Zusätzlich werden zu jedem Datensatz noch der TCA und der Titel angezeigt.

Das Langformat. Wenn der Benutzer einen Datensatz im Titelformat anklickt, dann gelangt er zu der Ausgabe des Langformats, die die meisten Metadaten zu jedem einzelnen Datensatz liefert:

ID MN 2006-09-15 **TC** 12:10:41:01 **Datenbank:** Köln

Punkt Zwölf vom: 15.09.2006	Beitrag: 9	Dauer 00:01:45:20	Status: Fertig
Sendeform: Bericht	Tape-Nr.	Ton: ZK	Material: Digi-Beta
Redaktion: Michaela Johannsen, Silke Teichmann			Studio: Kiel

BRD/BIELEFELD+WÜRZBURG: FALLBEISPIEL - MÜTTER VON MISSHANDEL TEN KINDERN TREFFEN SICH

ZFG: Nicole Gruhn darf sich keine Schwäche mehr erlauben, sie muss nämlich ihr Leben lang da sein für ihren behinderten Sohn Fabian (Behinderung erlitten durch heftiges Schütteln: **Schüttel-Trauma**). Der kam kerngesund zur Welt, wurde aber dann von seinem Vater, so schwer geschüttelt, dass er für immer ein Pflegefall bleiben wird. Wir haben über den Prozess berichtet. Das sah Silvia Amelung aus Bielefeld bei Punkt 12. Sie hat das gleiche Schicksal durchgemacht und bot Nicole spontan ihre Hilfe an. Jetzt haben die beiden Mütter sich zum ersten Mal getroffen. **!!ACHTUNG:** Die Mutter wird sowohl "Gruhn" als auch "Gruhn" geschrieben!! (vgl. FN 2006-08-29; N 2006-08-30; MN 2006-08-30)

DE: TREFFEN, MUTTER, VERGANGENHEIT, SCHÜTTELN, BABY, MISSHANDLUNG, UMGANG, BEWÄLTIGUNG, KIND
Diese Rechteangaben sind für den Bereich RTLInfo gültig!

TC D Herk	Video	Drt + Datum	Topic	Sequenz
12:10:09:24 31 sec.				Anmod
12:10:41:01 10 sec. RTL HAN fret		BRD / Bielefeld	Teiser, dumpfer Ton!	Abwärtsschwenk über Janek Amelung (behindert durch starkes Schütteln von Vater)
12:10:51:23 12 sec. RTL HAN fret		BRD / Bielefeld	Chyronfehler!	OT Silvia Amelung, Mutter (auf dem Chyronexemplar wird sie Nicole Gruhn genannt!) , O-Ton: Vater hat ihn massiv und wiederholt geschüttelt, der Vater kannte die Konsequenzen, der Vater hat den Tod des Kindes billigend in Kauf genommen, das Gericht urteilte auch so
12:11:04:21 5 sec. RTL HAN fret		BRD / Bielefeld	*	Janek Amelung spielt mit Mutter Nicole
12:11:10:19			Archiv, Gerichtssaal, vgl. FN 2006-08-	

Abbildung 3-4: Langformat TRIPHighway

An dieser Ausgabe erkennt man die Einteilung in Beitrags- und Sequenz-Metainformationen (vgl. Kapitel 2.3 „Wie kann man Metadaten unterteilen?“). Im obersten Teil der Anzeige [1] werden fast alle technischen Beitrags-Metadaten innerhalb einer Tabellenstruktur visualisiert.

Daraufhin folgen die drei größeren semantischen Metadatenfelder Zusammenfassung, Deskriptor (wörtlich: Beschreibung, hier: Schlagwort) und falls vorhanden Nebenbemerkung [2]. Die Zusammenfassung beinhaltet, wie der Name schon sagt, eine Kurzfassung des Beitrags. Das Feld Deskriptor enthält alle Schlagworte, die den

Beitrag inhaltlich charakterisieren. Die Nebenbemerkung dient für zusätzliche oder besondere Informationen (früher auch Rechtelage).

Die Kernkomponente des Ausgabeformats stellt eine Tabelle dar, die alle Metadaten zu jeder einzelnen Sequenz eines Beitrags enthält [3].

In der ersten Spalte sind der Timecode (*TC*), die Sequenzdauer „D“ (für Duration, engl.: Dauer) und die Herkunft der Sequenz dargestellt. Die Herkunftsinformation ist nur in den Beiträgen enthalten, die ab dem 23. Mai 2004 archiviert wurden. Vorher wurde diese Information im Feld Topic (engl.: Thematik) oder im Nebenbemerkungsfeld gespeichert. Seit November 2006 gibt es eine eigene Herkunftsdatenbank, auf deren Grundlage das farbliche Ampelsystem³⁰ dargestellt wird.

In der zweiten Spalte folgt ein Vorschaubild (Previewbild), wenn der Beitrag nach Dezember 1999 archiviert wurde. Ansonsten bleibt diese Spalte leer.

In der darauffolgenden Spalte folgen Metadaten bzgl. Ort und Datum. Der Ort schließt das Land mit ein und stellt die Informationen durch Trennzeichen hintereinander dar (Bsp.: Deutschland/Köln). Das Datum wird heutzutage kaum noch verwendet; wenn es gefüllt ist, beinhaltet es das Herstellungsdatum der Sequenz.

Die Spalte mit der Überschrift „Topic“ enthält eine kurze, thematisch charakterisierende Information zu der einzelnen Sequenz (Bsp.: Gericht, Staatsbesuch, Foto, Auf-sager, Animation usw.).

Die letzte Spalte dieser Tabelle enthält eine inhaltliche Beschreibung der Sequenz.

Die Fußzeile der oben beschriebenen Tabelle enthält noch Metadaten, die den Namen des Mediendokumentars, das Erfassungs- und Bearbeitungsdatum und die eindeutige Beitrags-*ID* beinhalten.

³⁰ weitere Informationen im Kapitel 6.1.1 „Ampelsystem“

Das Sequenzformat. Dieses Ausgabeformat zeigt alle Suchergebnisse an, die durch das Highlighting³¹-Verfahren von *TRIP* zurückgegeben werden. Das Highlighting-Verfahren sucht nach denjenigen Sequenzen eines Beitrags, die mindestens einen der gesuchten Volltext-Begriffe beinhalten.



Abbildung 3-5: Sequenzformat *TRIP*Highway

Visualisiert werden dann eine Mischform der Metadaten aus dem Titel- und Langformat. Diese sind durch eine Titelzeile [1] und eine Tabelle pro Beitrag mit den zutreffenden Sequenzen [2] strukturiert.

Die Titelzeile enthält die Informationen *ID*, Ort und Titel (siehe Titelformat *TRIP*-Highway).

Die angezeigte Tabelle besteht ebenfalls aus fünf Spalten und enthält die gleichen Metadaten, wie die Tabelle im Langformat (siehe Langformat *TRIP*Highway). Der

³¹ engl. Hervorheben

einzigster Unterschied zwischen den Tabellen besteht darin, dass nur die Sequenzen angezeigt werden, die die gesuchten Begriffe aus der Suchanfrage enthalten.

Das Videoformat. Diese Suchergebnisanzeige dient dem schnellen Zugriff auf die hinterlegten Videos und zeigt nur die Vorschaubilder der gefundenen Sequenzen an:



Abbildung 3-6: Videoformat TRIPHighway

Auf der einen Seite sind die darin enthaltenen Metadaten ein etwas größeres Vorschaubild und eine Bildunterzeile, welche die ID der Nachrichtensendung und die numerische Kennung der Sequenz beinhaltet.

Auf der anderen Seite werden die Metadaten ID, Topic und Zusammenfassung in Form eines sogenannten „Tooltips“ [1] zusätzlich angezeigt, wenn der Benutzer die Maus über das Vorschaubild bewegt.

Das Textformat. Dieses Ausgabeformat dient dem Ausdruck des Suchergebnisses.

Datenbank:NEWS_ARCHIV Treffer:1 von 3

ID MN 2006-09-15 TC 12:10:41:01 Datenbank: Köln

Punkt Zwölf vom: 15.09.2006 Beitrag: 9 Dauer 00:01:45:20 Status: Fertig
 Sendeform: Bericht Tape-Nr.: Ton: ZK Material: Digi-Beta
 Redaktion: Michaela Johannsen, Silke Teichmann Studio: Kiel

BRD/BIELEFELD+WÜRZBURG: FALLBEISPIEL - MÜTTER VON MISSHANDEL TEN KINDERN TREFFEN SICH

ZFG: Nicole Gruhn darf sich keine Schwäche mehr erlauben, sie muss nämlich ihr Leben lang da sein für ihren behinderten Sohn Fabian (Behinderung erlitten durch heftiges Schütteln: Schüttel-Trauma). Der kam kerngesund zur Welt, wurde aber dann von seinem Vater, so schwer geschüttelt, dass er für immer ein Pflegefall bleiben wird. Wir haben über den Prozess berichtet. Das sah Silvia Amelung aus Bielefeld bei Punkt 12. Sie hat das gleiche Schicksal durchgemacht und bot Nicole spontan ihre Hilfe an. Jetzt haben die beiden Mütter sich zum ersten Mal getroffen. !!!ACHTUNG: Die Mütter wird sowohl "Gruhn" als auch "Gruhn" geschrieben!! (vgl. FN 2006-08-29; N 2006-08-30; MN 2006-08-30)

DE: TREFFEN, MUTTER, VERGANGENHEIT, SCHÜTTELN, BABY, MISSHANDLUNG, UMGANG, BEWÄLTIGUNG, KIND
 Diese Rechteangaben sind für den Bereich RTLInfo gültig!

TC + Dauer	Herkunft + Rechte	Ort + Datum	Topic	Sequenz
12:10:09:24 31 sec.				Anmod
12:10:41:01 10 sec.	RTL HAN freet	BRD / Bielefeld	Teisen, dumpfer Ton!	Abwärtsschwenk über Janek Amelung (behindert durch starkes Schütteln von Vater)
12:10:51:23 12 sec.	RTL HAN freet	BRD / Bielefeld	Chyronfehler!	OT Silvia Amelung, Mutter (auf dem Chyronskript wird sie Nicole Gruhn genannt!) ; 0-Ton: Vater hat ihn massiv und wiederholt geschüttelt, der Vater konnte die Konsequenzen, der Vater hat den Tod des Kindes billigend in Kauf genommen, das Gericht urteilte auch so
12:11:04:21 5 sec.	RTL HAN freet	BRD / Bielefeld	*	Janek Amelung spielt mit Mutter Nicole
12:11:10:19 4 sec.	RTL LPZ freet	BRD / Leipzig	Archiv, Gerichtssaal, vgl. FN 2006-08-29 "BRD: BABY VOM EIGENEN VATER SCHWER MISSHANDELT"	Zoom auf Anklagebank mit Mann Rene G. (Vater)
12:11:16:11 3 sec.	RTL LPZ freet	BRD / Leipzig	Archiv, Gerichtssaal	Täter Rene G. (seitl.) in Gerichtssaal
12:11:18:13 3 sec.	RTL LPZ freet	BRD / Leipzig	Archiv, Foto	Mutter Nicole Gruhn (Gruhn) mit Sohn Fabian (anonymisiert)
12:11:21:18 8 sec.	RTL HAN freet	BRD / Bielefeld	*	Silvia Amelung steigt mit Sohn Janek in PKW ein

Abbildung 3-7: Textformat TRIPHighway

Hier werden alle Metadaten tabellarisch aufbereitet, die auch schon im Langformat beschrieben wurden.

Der einzige Unterschied ist, dass die 2. Spalte der Sequenz-Tabelle kein Vorschaubild der Sequenz, sondern die Information über Rechte und Herkunft enthält, die in dem Langformat mit in der 1. Spalte untergebracht sind.

3.1.3 Weiterverarbeitung

Sobald der Redakteur ein Suchergebnis in die nähere Auswahl nimmt und nachschauen möchte, ob das Bewegtbild-Material seinen Ansprüchen gerecht wird (indem es die gesuchte Person, Szene oder den gesuchten Ort enthält), kann er sich das Videomaterial mit Hilfe der Software VPMS PreviewClient³² anschauen.

Hierzu klickt er in den Ausgabeformaten, die eine Vorschau ermöglichen, auf das gewünschte Previewbild der Sequenz. Dadurch wird eine Funktion aufgerufen, die

³² weitere Informationen im Kapitel 6.1.3 „VPMS PreviewClient“

eine Schnittstelle des Videoplayers aktiviert und die notwendigen Parameter übergibt, um die Videosequenz eindeutig zu identifizieren und zu starten.

Damit ist der Funktionsumfang von *TRIP*Highway klar abgegrenzt: Die webbasierte Anwendung ermöglicht lediglich die Recherche nach Archivmaterial, während die Sichtung, Vorerfassung und der Schnitt des Bewegtbild-Materials über die Software *VPMS* PreviewClient (siehe Abbildung 6-2) und die Erfassung des zu archivierenden Materials durch den *TRIP* Client (siehe Abbildung 2-3) durchgeführt werden.

3.2 Probleme und Engpässe

Anhand des oben schon verwendeten Beispiels „Schüttel-Trauma“ werden die drei negativen Kernpunkte erläutert, die eine Optimierung der Anwendung *TRIP*Highway notwendig machen.

3.2.1 Aufwand

Die Archive werden aus firmenpolitischen Gründen bisher getrennt verwaltet. Aus diesem Grund kann der Redakteur momentan nur nach allen archivierten Beiträgen aus dem Newsarchiv recherchieren.

Alle von RTL Creation produzierten, grafischen Elemente wie Animationen, Grafiken oder Texttafeln (siehe Abbildung 3-8) sind als Rohmaterial im Creationarchiv gelagert. *Nur* die Mediendokumentare von RTL Creation können diese Materialien in bestimmten Ausgabeformaten (analog zu denen im Newsarchiv – siehe Kapitel 3.1.2) durch die Anwendung *TRIP* Client (siehe Kapitel 2.5.1) wieder auffinden.



Abbildung 3-8: produzierte Elemente von RTL Creation

Zusätzlich zur Speicherung im Creationarchiv werden die grafischen Elemente durch die Sendungsaufzeichnung der Nachrichten aber zusätzlich in *ausgestrahlter* Form im Newsarchiv archiviert.

Momentan kann der Redakteur diese ausgestrahlten Sequenzen im Newsarchiv mit Hilfe von *TRIPHighway* wieder auffinden und nur unverändert in den neuen Beitrag integrieren.

Ein großer Nachteil bei diesem Arbeitsschritt ist, dass der Redakteur nicht wissen kann, ob sich die Designvorlage der Grafik/Animation in der Zwischenzeit verändert hat oder ob die Rohvorlage der Sequenz umfangreicher ist und er eigentlich noch mehr Informationen in dem Beitrag unterbringen könnte.

Ein gutes Beispiel für diese Problematik ist die Schüttel-Trauma Animation (siehe erste Grafik Abbildung 3-8): Der Bildhintergrund ist in einem lila-orangen Farbverlauf gefärbt, welcher die Designvorlage einer Nachrichtensendung bis Mitte 2004 kennzeichnete. Die Animation wurde in dieser Form archiviert und Ende August 2006 von einem Rechercheur durch *TRIPHighway* wieder gefunden. Der Suchende integrierte die gefundene Schüttel-Trauma Animation mit veralteter Formatvorlage in seinen Beitrag, der kurze Zeit später ausgestrahlt wurde.

Die Imageverluste durch Einfügen solcher fehlerhaften Animationen oder Grafiken für eine Nachrichtensendung und für die produzierende Firma RTL Creation ist enorm. Gleichzeitig schwächt es die Arbeitsqualität des Redakteurs.

Bisher kann der Rechercheur die Aktualität und Korrektheit der Animation bzw. Grafik nur feststellen, wenn er sich telefonisch an einen TV-Designer oder einen Mediendokumentar von RTL Creation wendet.

Zusätzlich kann er das Grafik- oder Animations-Rohmaterial, welches im Creationarchiv gelagert ist, selber nicht online einsehen oder in seinen neuen Beitrag integrieren.

Der Rechercheaufwand des Redakteurs ist in einem solchen Fall erheblich und nimmt viel Zeit in Anspruch.

3.2.2 Inflexibilität

Das serverseitige Modul - ebenfalls mit dem Namen *TRIPHighway* (siehe Abbildung 2-3) - ist von der Firma TietoEnator implementiert und verursacht folgende Probleme.

- Die *CBC* ist vom Hersteller abhängig, da auftretende Fehler nur von der Firma TietoEnator behoben werden können und dieser Vorgang zeitaufwendig ist.
- Die Komponente kann nur zu Recherchezwecken in eine Webanwendung integriert werden. Eine Dokumentation und Erfassung von Archivmaterial ist nicht möglich, da keine Funktionen vorhanden sind, die den Schreibzugriff auf die Datenbank *TRIP* ermöglichen. Ein solches Modul wurde im Jahr 2000 von der Firma TietoEnator unter dem Namen *TRIPHighwayPro* angekündigt, jedoch nie auf den Markt gebracht.
- Die Software beinhaltet keinerlei Programmierschnittstellen und kann nicht funktional erweitert werden. Die daraus resultierende Inflexibilität erlaubt keinerlei Änderungen in der kompletten *TRIPHighway*-Architektur.

3.2.3 Kosten

Das serverseitige Modul *TRIPHighway* verursacht sowohl Lizenz- als auch dauerhafte Wartungskosten.

3.3 Rahmenbedingungen

Bevor die Anforderungen umgesetzt werden, müssen noch folgende Rahmenbedingungen beachtet werden, die der Programmierer zwingend einhalten soll.

1. Das äußere Erscheinungsbild (Design) der Anwendung sollte möglichst wenig verändert werden, um den Benutzer nicht zu verwirren. Der Arbeitsaufwand und die dadurch verursachten Kosten, die eine notwendige Neueinarbeitung auslösen würde, wären immens.
2. Die Zugriffszeiten der bisher möglichen Suche sollten unter der funktionalen Erweiterung keinesfalls steigen. Bisher benötigt eine Standardsuche, in der ein

bis zwei Suchfelder gefüllt sind, weniger als 5 Sekunden. Jede Suche, die länger als 30 Sekunden dauert, wird automatisch von der Webanwendung beendet.

3. Die Metadaten-Ausgabe in die einzelnen *TRIP*Highway Formate sollte so wenig wie möglich geändert werden, da dann ein erheblicher Anpassungsaufwand bei den serverbasierenden Ausgabeformaten entsteht.

3.4 Lösungsansätze

3.4.1 Erweiterung der Recherchemöglichkeit

In Zukunft soll schnittstellenübergreifende Recherche zwischen dem Newsarchiv und Creationarchiv möglich sein, damit die genannten Probleme (siehe Kapitel 3.2.1) beseitigt werden.

Es gibt drei verschiedene Lösungsansätze, welche die gleiche Funktionalität umsetzen, aber unterschiedliche Auswirkungen auf die schon bestehenden Funktionalitäten oder auf das Anwendungsdesign haben.

Gleichzeitige Suche über beide Archivsysteme. Bei diesem Ansatz wird der Umgang mit der Suchmaske (siehe Abbildung 3-1) nicht verändert. Allerdings wird beim Starten der Suche durch Klick auf den Button „Suche“ oder durch Betätigen der „Enter“-Taste die Suchanfrage sowohl an das Newsarchiv, wie bisher, als auch an das Creationarchiv gestellt.

Das Suchergebnis des Creationarchivs wird mit dem des Newsarchivs in einem Ausgabeformat visualisiert und vergleichend nebeneinander dargestellt.

Vorteile	Nachteile
keine Änderungen an der Suchmaske nötig	höhere Wartezeit für den Benutzer durch Zugriff auf zwei Datenbanken
Recherche über beide Archivsysteme ohne zusätzlichen Aufwand für den Benutzer	unübersichtliche Ergebnisdarstellung bei größerer Trefferanzahl

Abbildung 3-9: Vor- und Nachteile Lösungsansatz 1

Separate Suchmöglichkeit im Creationarchiv. Die Idee hinter diesem Ansatz ist, das bisherige *TRIP* Highway gar nicht zu verändern, sondern nur zu erweitern. Es wird eine spezielle Suchmaske für die Felder des RTL Creation Archivs erschaffen, die jederzeit vollkommen getrennt von der Newsarchivsuche ausgeführt werden kann. So kann ein Rechercheur bei Bedarf das Rohmaterial völlig eigenständig in der Creation-Datenbank suchen.

Vorteile	Nachteile
Suchergebnispräsentation des Creationarchivs analog zur Ergebnisdarstellung im Newsarchiv	lose Kopplung zwischen den beiden Archivsystemen
Rechercheur kann das Material im Creationarchiv eigenständig durchsuchen	Vergleich des Rohmaterials und des ausgestrahlten Materials muss vom Rechercheur durchgeführt werden
keine Performanceeinbußen bei der Suche	Benutzer muss grundlegende Kenntnisse über die Formulierung von Suchanfragen speziell für das Creationarchiv haben, um die Originalmaterialien zu finden

Abbildung 3-10: Vor- und Nachteile Lösungsansatz 2

Suchanfrage im Creationarchiv auf Wunsch. Bei diesem Ansatz wird die herkömmliche Recherche nicht verändert. Das Suchergebnis wird im bisherigen Ausgabeformat präsentiert, ist die gesuchte Sequenz ein von RTL Creation produziertes Medium und somit auch in der Creationarchiv Datenbank enthalten, so erscheint in diesem Datensatz ein Link, der es ermöglicht, das Rohmaterial dieser Sequenz in der Creationarchiv-Datenbank zu suchen und in einem separaten Ausgabeformat anzuzeigen. Sollte das Rohmaterial nicht vorhanden sein, wird eine Fehlermeldung angezeigt.

Vorteile	Nachteile
keine Änderungen an der Suchmaske und wenig Änderungen an den Ausgabeformaten nötig	höhere Wartezeit für den Benutzer bei der Präsentation des Suchergebnisses

Redakteur benötigt keine Kenntnisse über die Creation-Datenbank.	wenn die Suche im Creationarchiv erfolglos ist, kann der Redakteur die Suche nicht selbstständig ausführen
Rechercheur muss die Suchanfrage an das Creationarchiv nicht selbstständig formulieren.	

Abbildung 3-11: Vor- und Nachteile Lösungsansatz 3

Alle drei Lösungsansätze haben ihre Schwächen und Stärken. Um das beste Ergebnis zu erreichen, darf weder der Rechercheaufwand gesteigert, noch die Performance beim Zugriff verschlechtert werden.

Damit die beiden wichtigen Ziele erfüllt werden, wird eine Kombination aus den obigen Lösungsansätzen gewählt: Um die Geschwindigkeit beim Zugriff auf die Datenbank nicht zu verschlechtern, wird die Suche nicht parallel durchgeführt, sondern je nach Bedarf. Da der Rechercheaufwand und die damit verbundene Einarbeitung nicht steigen soll, muss dem Suchenden die Chance gegeben werden, die Suchanfrage an die Creations-Datenbank abzusetzen ohne die Suchanfrage selber zu formulieren. Diese Argumente sprechen für den Lösungsansatz 3.

Der fehlende Einfluss des Rechercheurs bei der Auffindung des Rohmaterials aus dem Creationarchiv ist das Problem bei diesem Verfahren. Sollte keine passende, archivierte Sequenz gefunden werden, muss der Benutzer noch die Chance haben, die vom Programm ausgeführte Suche anzupassen und mit Hilfe von veränderten Suchkriterien doch noch das Original aufzufinden, auch das wird somit Teil der Soll-Anforderung.

Eine weitere Schwierigkeit ist die übersichtliche Darstellung der Rohmaterialien im Trefferfall. Da die gefundenen Materialien nur in einem neuen Ausgabeformat dargestellt werden können, müssen sie trotzdem so präsentiert werden, dass ein Benutzer das zutreffende Rohmaterial aus dem Creationarchiv direkt erkennt und dem entsprechenden schon verwendeten Material aus dem Newsarchiv zuordnen kann.

Die Umsetzung dieser entwickelten Anforderungen wird in Kapitel 4 beschrieben, dabei werden auch aufgetretene Probleme und Anpassungen erläutert.

3.4.2 Neuentwicklung des Servermoduls

Damit die Recherche auch langfristig noch über die Webanwendung durchgeführt werden kann, selbst wenn sich die Datenbank- oder andere Servertechnologien ändern, muss die zentrale Serverkomponente *TRIPHighway* neu entwickelt werden.

Weitere Beweggründe für die Neuentwicklung sind die jährlich einzusparenden Kosten und die Herstellerunabhängigkeit.

Nachfolgend werden drei verschiedene Lösungsansätze vorgestellt, die alle die Ablösung der jetzigen Serverkomponente vorsehen, allerdings durch die unterschiedlich eingesetzten Programmiersprachen und Techniken, verschiedene Vorteile bieten.

Entwicklung des Servermoduls mit der Programmiersprache Java³³. Dieser Ansatz beinhaltet eine komplette Neuentwicklung aller Funktionalitäten und Klassen des jetzigen *TRIPHighways*. Zum Zugriff auf die Volltextdatenbank kann eine Java-API der Firma TietoEnator verwendet werden.

Vorteile	Nachteile
Plattformunabhängigkeit	evtl. Kosten durch Hardwareerweiterung, damit die gleiche Performance erreicht wird
hohe Anzahl bereits zur Verfügung stehender Klassen und Methoden für Standardoperationen durch Java-APIs der Firma SUN	Java-API der Firma TietoEnator bietet nur eingeschränkten Funktionsumfang beim Zugriff auf die <i>TRIP</i> -Datenbank
geringer Einarbeitungsaufwand	

Abbildung 3-12: Servermodul Vor- und Nachteile Lösungsansatz 1

Umsetzung der Komponente in der Skriptsprache PHP³⁴. Eine weitere Möglichkeit wäre die Entwicklung mit Hilfe von *PHP*. Hierbei müsste zunächst eine Schnittstelle zur *TRIP-API* erschaffen werden, um Zugriff auf die *TRIP*-Datenbank zu erhalten.

³³ weitere Informationen im Kapitel 6.3.5 „Java“

³⁴ Hypertext Preprocessor

Vorteile	Nachteile
<i>PHP</i> kann die gleichen Performance-werte wie die momentan eingesetzte Lösung erreichen	Mehraufwand durch Schnittstellenprogrammierung zur <i>TRIP-API</i>
	hoher Einarbeitungsaufwand

Abbildung 3-13: Servermodul Vor- und Nachteile Lösungsansatz 2

Implementierung der Neuentwicklung mit Hilfe von C++³⁵. Dieses Vorgehen beinhaltet ebenfalls die komplette Neuentwicklung des Servermoduls, allerdings fällt die Schnittstellenprogrammierung, durch Zugriff auf die in C geschriebene, bereits vorhandene *TRIP-API*, weg.

Vorteile	Nachteile
C++ kann die gleichen Performance-werte wie die momentan eingesetzte Lösung erreichen	hoher Einarbeitungsaufwand
weniger Arbeitsaufwand durch Einbindung der <i>TRIP-API</i>	
Viele Grundfunktionen sind in C++ bereits implementiert, da das Projektteam C++ als Standardsprache einsetzt	
geringer Wartungsaufwand	

Abbildung 3-14: Servermodul Vor- und Nachteile Lösungsansatz 3

Die Entscheidung fällt aufgrund der vielen ausschlaggebenden Vorteile für den Lösungsansatz 3.

Neben den hohen Performanceansprüchen, die im Gegensatz zum 1. Lösungsansatz, ohne zusätzlich entstehende Kosten erfüllt werden können, ist die langjährige Erfahrung des Projektteams mit der objektorientierten Sprache C++ ein weiterer großer Einflussfaktor.

³⁵ Weitere Informationen im Kapitel 6.3.1 „C++“

Da man bedenken muss, dass das entwickelte Servermodul zukünftig gewartet und eventuell auch erweitert werden soll, ist der Arbeitsaufwand bei dieser Lösung sehr viel geringer, als wenn die Implementierung in Java erfolgen würde.

Die Skriptsprache *PHP* wird nicht eingesetzt, da ein hoher Mehraufwand bei der Implementierung der Schnittstelle zur *TRIP-API* entsteht.

Der nachteilige hohe Einarbeitungsaufwand, der bei der Umsetzung des neuen *TRIPHighways* mit C++ entsteht, kann durch die Unterstützung der anderen Teammitglieder kompensiert werden.

4 Umsetzung

Die Realisierung der archivübergreifenden Recherche, die Entwicklung eines Bildvergleichs zur Hervorhebung der gesuchten Sequenz und die Portierung des Servermoduls sind Bestandteil der Umsetzung. Die Ergebnisse werden anhand von praktischen Beispielen präsentiert und erläutert.

4.1 Archivübergreifende Recherche

Damit die in den Lösungsansätzen (siehe Kapitel 3.4.1) definierten Ziele erreicht werden, müssen sowohl die clientseitigen Implementierungen, als auch die serverseitigen Implementierungen angepasst werden. Die nachfolgend beschriebenen Änderungen bzw. Erweiterungen habe ich im Rahmen der Bachelorarbeit umgesetzt.

4.1.1 Optimierung der Webanwendung

Zuerst legte das Projektteam zusammen mit den Auftraggebern des RTL Creationarchivs fest, welches der fünf Ausgabeformate (siehe Kapitel 3.1.2) einen Verweis, in der jeweiligen Sequenz, zum Creationarchiv enthalten soll. Hier boten sich generell nur die zwei Formate Lang und Sequenz an, da die Sequenz-Metadaten in einer Tabellenstruktur angeordnet sind und die Formate die Previewbilder der jeweiligen Sequenzen beinhalten.

Innerhalb der zwei in Frage kommenden Ausgabeformate wurde als nächstes festgelegt, wo der Verweis zum Creationarchiv angezeigt werden soll.

TC D Herk	Video	Ort + Datum	Topic	Sequenz
12:00:10:15 10 sec. PB ■ frei		* / *	animierte Grafik	Schüttel-Trauma.

Abbildung 4-1: Auswahl des Feldes im Ausgabeformat

Da der Link eine Suchanfrage anstoßen wird, die sich auf die einzelne, zu suchende Sequenz bezieht, muss der Verweis in die Tabellenstruktur der einzelnen Ausgabeformate integriert werden.

Eine Möglichkeit war, ein neues Tabellenfeld einzuführen, welches den Verweis enthält. Allerdings hätte dies eine Änderung der Tabellenstruktur der Ausgabeforma-

te zu Folge und würde damit zu Unübersichtlichkeit und Verwirrung des Anwenders führen.

Eine Alternative dazu war die Integration des Links in ein schon vorhandenes Feld. Diese Variante wurde gewählt; dabei wurde der Verweis für die Entwicklungs- und Testphase im Feld Topic eingefügt (siehe Abbildung 4-1). Hier wurde dem Benutzer bisher nur eine thematische Einordnung der Sequenz angezeigt, so dass ein zusätzlicher Hinweis in Form eines Verweises nicht stört. Der Suchende kann somit darauf hingewiesen werden, dass er bei der Anzeige des Links die Möglichkeit hat, das Rohmaterial der thematisch gleichen Sequenz im Creationarchiv zu finden.

Nachdem die Platzierung des Verweises festgelegt war, erfolgte die Anpassung der Ausgabeformate Lang und Sequenz. Im Folgenden wird ein Auszug der entscheidenden Codestelle dargestellt (siehe Abbildung 4-2) und erläutert:

```

<box <t=_<table border="0" id="tblCreation" _>
    1
    _<tr_>
    _<td id="content1" _>>>
    <box <if-nonempty(TOPIC.x) > TOPIC.x > 2
    <box <t=_</td_></tr_><tr_><td id="content2" _>>>
    <box Herkunft.x> 3
    <box <t=_</td_></tr_><tr_><td id="content3" _>>>
    <box <t=_<br_><a href="javascript:CallCreation('>ID<t=');
    " _> _<i_> Suche Rohmaterial Grafik _</i_></a_> > >
    4
</box <t=_</td_></tr_></table_>>>

```

Abbildung 4-2: Codeauszug Erzeugung des Links

Wie man erkennen kann erscheint hier eine etwas andere Syntax, als die Auszeichnungssprache *HTML*. Dies liegt daran, dass der *TRIP*-Hersteller TietoEnator Schlüsselzeichen und Begriffe in einer eigenen Sprache namens OutputForm (Dateiendung „.OF“) definiert hat, welche es *TRIP* ermöglicht, dynamische Ausgabeformate zu generieren. Die oben verwendeten *TRIP* spezifischen Befehle sind:

- „<“ und „>“ kennzeichnen den Anfang und das Ende eines Blocks. Jeder Block kann eine beliebige Anzahl an Boxen, Text, Funktionsaufrufen oder Feldnamen beinhalten.
- „t=“ markiert den Anfang eines Textes innerhalb eines Blocks
- „/“ symbolisiert einen Zeilenumbruch in der Ausgabe
- „if-nonempty“ steht für eine Bedingung, welche den Feldinhalt überprüft
- „_“ ist ein reserviertes Zeichen, um Sonderzeichen (wie z.B. „/“, „<“, „>“) explizit als HTML zu kennzeichnen. So wird „<TABLE>“ in einem Block des Ausgabeformats als „_<TABLE_>“ geschrieben.

Der oben dargestellte Code definiert eine Tabelle im Topicfeld mit der *ID* „tblCreation“ [1] und 3 Zeilen, wovon die erste Zeile das Thema [2], die zweite Zeile die Herkunft [3] und die dritte Zeile den Link [4] beinhaltet.

In der fertig aufbereiteten Ausgabe würde das Ergebnis nach den serverseitigen Änderungen so aussehen:

12:00:04:03 6 sec. RTL LPZ ■ frei		BRD / Leipzig	* RTL LPZ <i>Suche Rohmaterial Grafik</i>	OT Nicole Grun, Mut
12:00:10:15 10 sec. PB ■ frei		* / *	animierte Grafik PB <i>Suche Rohmaterial Grafik</i>	Schüttel-Trauma.
12:00:21:00 8 sec. RTL LPZ ■ frei		BRD / Leipzig	Foto RTL LPZ <i>Suche Rohmaterial Grafik</i>	Sohn Fabian (anonym)

Abbildung 4-3: Ausgabe des Langformats vor Formatierung

Damit der Link nicht (wie in Abbildung 4-3 dargestellt) in jedem Topicfeld angezeigt wird, sondern nur in den relevanten Sequenzen, die von RTL Creation hergestelltes Material aus dem Bereich Information beinhalten, wurde hierfür eine Logik in Form einer Javascript-Funktion implementiert, die nach bestimmten Schlüsselwörtern sucht.

Nach Auskunft des Newsarchivs werden die ausgestrahlten Sequenzen, die von RTL Creation produziert wurden, seit Mitte 2004 immer mit dem Kürzel „PB“ (für Paintbox) im Herkunftsfeld gekennzeichnet. Für ältere Archivbeiträge gilt jedes Schlüs-

selwort im Topicfeld, welches die Buchstabenfolge „anim“ enthält. Dies kann eine animierte Grafik, eine animierte Karte, eine Animation oder Ähnliches sein.

Sobald der Browser die Javascript-Funktion ausgeführt hat, wird folgende Darstellung präsentiert:

12:00:04:03 6 sec. RTL LPZ ■ frei		BRD / Leipzig	*	OT Nicole Grün, Mü
12:00:10:15 10 sec. PB ■ frei		* / *	animierte Grafik Suche Rohmaterial Grafik	Schüttel-Trauma.
12:00:21:00 8 sec. RTL LPZ ■ frei		BRD / Leipzig	Foto	Sohn Fabian (anonym)

Abbildung 4-4: Ausgabe des Langformats nach Formatierung

Automatische Suchanfrage generieren. Wenn der Rechercheur in einem der beiden Ausgabeformate auf den Link „Suche Rohmaterial Grafik“ klickt, soll eine Suchanfrage an die Datenbank des RTL Creationarchiv gestellt werden, die den passenden Datensatz findet. Hierzu sind mehrere Schritte notwendig:

- Bestimmung der eindeutigen Identität der Sendung
- Syntax-Analyse der Identität
- Formulierung und Übersetzung der Suchanfrage
- Visualisierung des Suchergebnisses

Die eindeutige Identität einer Sendung wird durch das Schlüsselfeld *ID* bestimmt.

Beispiel *ID* = MN 2006-08-30

Die ersten beiden Buchstaben enthalten immer das Kürzel der Sendung (vgl. Abbildung 3-3) und danach folgt, durch ein Leerzeichen getrennt, das Ausstrahlungsdatum in amerikanischer Notation. Diese *ID* wird nun einer Javascript-Funktion übergeben, die ausgelöst wird, sobald der Verweis angeklickt wurde. Sie zerlegt die *ID* in die einzelnen Teile und formuliert die Suchanfrage.

Allerdings kann die Suchanfrage nicht sofort abgesetzt werden, da das Creationarchiv sowohl andere Sendungskürzel, als auch eine andere Datumsnotation (deutsches Format) in der Datenbank speichert.

Sendungskürzel Newsarchiv	Sendungskürzel Creationarchiv
EE	Extra
EL	Life
ET	EX
EX	Explo
FN	P6
FN9	P9
Form	-
Fuba	-
MN	P12
N	HN
S-N	HN Spezial
NJ	NJ
S-NJ	NJ Spezial
Ski	-
SN	-

Abbildung 4-5: Vergleich Sendungskürzel

Um dieses Problem zu beheben folgt zuerst ein Sendungskürzel-Vergleich (siehe Abbildung 4-5). Ist der Vergleich erfolgreich, wird das Kürzel ausgetauscht. Ist das passende Sendungskürzel im Creationarchiv nicht vorhanden, wird eine Fehlermeldung ausgegeben und die Suchanfrage wird nicht ausgeführt (siehe Abbildung 4-6).

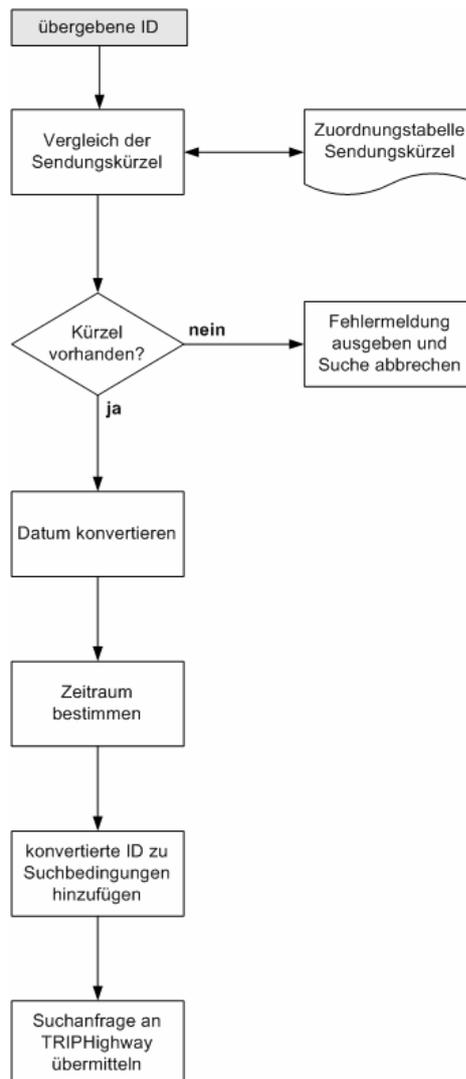


Abbildung 4-6: Workflow "Suchanfrage generieren"

Im Erfolgsfall folgt die Konvertierung des Datums in das deutsche Format. Zusätzlich wird die Suchanfrage so formuliert, dass ebenfalls über einen Tag früher und später gesucht wird. Die Suche über den vorhergehenden Tag wird durchgeführt, da die Designer manchmal Grafiken/Animationen vorproduzieren, wenn ein Ereignis vorhersehbar ist (z.B. Ausstellung oder Jahrestag). Die Suche über den nachfolgenden Tag ist als zeitlicher Puffer gedacht, da die produzierten Materialien zwar zeitnah, manchmal aber erst einen Tag später archiviert werden.

Schließlich wird die Suchanfrage per *HTML-GET-Request* über die *CGI* an das Servermodul *TRIPHighway* übergeben. In dieser *URL* sind alle datenbankspezifischen Informationen enthalten, die *TRIPHighway* für die Suchanfrage benötigt (siehe Kapitel 4.3.1).

Zuletzt wird das von *TRIPHighway* zurückgelieferte Suchergebnis in Form einer *THW*-Datei im Browser visualisiert. Hier hilft dem Entwickler die Unterteilung der Webanwendung in drei Frames (engl.: Rahmen):



Abbildung 4-7: Frameunterteilung *TRIPHighway*

Der oberste Frame „frameNav“ beinhaltet alle Navigationselemente und zu speichernde Daten. Er bleibt bei jeder Ansicht der Webanwendung bestehen.

Der untere Frame „frameHelp“ ist eine Unterstützung für den Entwickler und bleibt dem Anwender gegenüber verborgen. Er versteckt die Rückmeldungen der Serverseite und dient dazu, serverseitige Fehlermeldungen abzufangen.

Der mittlere Frame „frameDoc“ ist das Kernstück und beinhaltet immer die für den Benutzer relevanten Daten. Hierzu gehören die Suchmaske, die Auswahl der Bereiche und der Datenbank und die Präsentation der Suchergebnisse in Form der Ausgabeformate. Der Inhalt dieses Rahmens wird je nach Benutzerinteraktion ausgetauscht, auch bei der Darstellung der *THW*-Datei. Hierzu muss dem Frame die Datei zugewiesen werden, die an dieser Stelle präsentiert werden soll.

In diesem Fall ist es das Suchergebnis des Creationarchivs, in dem die gewünschten Rohmaterialien enthalten sein sollten:

1 Treffer 1 von 2 [Back](#)

1 **2** Creationsarchiv Rechte Hilfe frameNav

ID: DGN 00055 TC: 01:38:35:06 Datenbank: CREATION

Animation vom: Beitrag: 114 Dauer: 00:01:49:05 Status: Fertig
 Sendeform: Bericht Tape-Nr: 00055 Ton: Kein Ton Material: Digi-Beta gross
 Sendedatum: Producer: Operator: Studio:

NEWS ARCHIV BAND NR. 55: KARLCHENS CLIPS, USA / HURRIKAN, NATASCHA, GALERIE KABINETT - BAUSTELLE IM BUNDESTAG

ZFG:
 2D-Animationen
 3D-Animationen
 3D-Stop-Animation
 Fotostillschüssiges Material:

NB: Bereinigt am 05.01.2007 um 15:34:29

Video	TC	Topic	Sequenz
	01:38:35:06	Animation	P12 - Karlchens Clips. (Readymade, Alle Tools + Keys), Karfchen, www.clipfish.de
	01:39:36:14	Karte	HI - USA / Hurrikan, Wirbelsturm zieht von Kuba nach Miami, weiße Punkte Linie, New Orleans.
	01:39:57:05	3D Animation	HI - Natascha Kampusch, Haus des Entführers Wolfgang Priklopil, Zoom Garage - Klappe, Fahrt in den Keller, Tür öffnet sich zum Verlies, Schwenk durch Zimmer mit Hochbett, Klo, Waschbecken, Regal, Schreibtisch, Bücher
	01:40:14:24	Animation	HI - Galerie Kabinett, Baustelle im Bundestag, Bundesadler, CDU/CSU, SPD, Wamschid, Gesundheitsreform, Arzt - Aeskulapstab, Arbeitsamt - Gang - Wartende, Deutschland - Terrorbekämpfung - Fadenkreuz, schwarz rot gold, Bundeswehreinsetzung, Soldat UNO

Dok: cpr ED: 12.09.2006 Bearb.: BD: 21.09.2006 frameDoc

frameHelp

Abbildung 4-8: Ausgabe des Suchergebnisses des Creationarchivs

Bei der Präsentation des Suchergebnisses im Lang- und Sequenzformat werden meistens drei Suchergebnisse gefunden, eins für jeden gesuchten Tag. Werden weniger Treffer gefunden, wurde an einem Tag kein Sammelband³⁶ angelegt. Gibt die Suchanfrage mehr Treffer zurück, sind die produzierten Materialien aus dem Bereich Information auf mehreren Sammelbändern aus Platzgründen verteilt archiviert worden. In diesem praktischen Beispiel des „Schüttel-Traumas“ im Anzeigeformat „Lang“ werden zwei Suchergebnisse gefunden [1]. Das Erste enthält vier produzierte Rohmaterialien vom 29.08.2006 (siehe Abbildung 4-8); das Zweite enthält sieben produzierte Rohmaterialien vom 30.08.2006. Da der Rechercheur meist unter Zeitdruck steht und man ihm jeglichen Aufwand ersparen sollte, entstand die Idee den gesuchten Datensatz hervorzuheben.

³⁶ Auf einem Sammelband werden alle produzierten Materialien von jeweils einer Informationssendung chronologisch, hintereinander archiviert.

Allerdings ist dies nicht über Schlüsselfelder, wie beispielsweise das Identitäts-Feld, möglich, da diese zwischen den Archivsystemen nicht übereinstimmen. Eine andere Möglichkeit wäre, den Volltext des Sequenzfeldes im Newsarchiv, in Teile zu zerlegen und im Sequenzfeld des Creationarchivs zu suchen (sogenanntes Text-Matching).

Allerdings hat diese Option wenige Erfolgsaussichten, da die Mediendokumentare des News- und Creationsarchiv nach unterschiedlichen, inhaltlichen Schwerpunkten dokumentieren.

Deswegen ist ein Bildvergleich der beiden archivierten Vorschaubilder die sinnvollste Möglichkeit. Dieser Ansatz wird im nächsten Kapitel (4.2) beschrieben.

Findet die Suchanfrage keine passenden Datensätze zur Identität der Sendung innerhalb des gesuchten Zeitraums, soll eine Fehlermeldung ausgegeben werden.

Da es vorkommen kann, dass der Datensatz im Creationarchiv vorhanden ist, aber nicht durch die automatisch generierte Suchanfrage aufgefunden wird, da die *ID* nicht übereinstimmt oder das Archivdatum nicht innerhalb des Zeitraums liegt, habe ich eine Qualitätssicherung durchgeführt. Bei 100 zufällig ausgewählten Suchanfragen an das Creationarchiv, wurden 11 nicht aufgefunden, obwohl sie im Creationarchiv vorlagen. Dies entspricht einer Trefferquote von 89 %. Um die restlichen Datensätze aufzufinden muss der Redakteur die Option haben, selbstständig - mit Hilfe einer Suchmaske – über das produzierte Material aus dem Bereich Information im Creationarchiv zu recherchieren.

Flexible Suche über das Creationarchiv. Die flexible Suche muss möglich sein, wenn kein passender Datensatz im Creationarchiv gefunden wurde oder wenn Datensätze gefunden wurden, die nicht das gewünschte Rohmaterial enthalten. Zur Erfüllung dieser Aufgabe wurde eine Suchmaske realisiert, welche die relevanten Felder beinhaltet und dem Design der Suchmaske aus dem Newsarchiv gleicht.

Abbildung 4-9: Suchmaske Creationarchiv

Wird kein Suchergebnis in der Volltextdatenbank gefunden, wird das *HTML* Formular (siehe Abbildung 4-9) automatisch in den Frame „frameDoc“ geladen. Ansonsten ist die Maske über einen Button in der Navigationsleiste zu erreichen, sobald die Suchergebnisse aus dem Creationarchiv dargestellt werden (siehe Abbildung 4-8 [2]).

Die Darstellung und Interaktion mit der Suchmaske ist aus Gründen der Gebrauchstauglichkeit (usability) analog zur Newsarchiv-Suchmaske. Es werden auch hier die Trefferanzahl der einzelnen Felder hinter dem Eingabefeld dargestellt und nach Betätigen der „Enter-Taste“ oder des Suchbuttons wird die Gesamt-Trefferanzahl berechnet.

In der Navigationsleiste erscheint oben links nur noch ein Ausgabeformat – das Langformat. Die Präsentation der Suchergebnisse in diesem Format wurde ausgewählt, da es die meisten Informationen über die gefundenen Sequenzen und über das Sammelband, auf dem die Sequenzen archiviert sind, enthält.

4.1.2 Schwierigkeiten bei der Implementierung

Als unvorhersehbares Problem stellten sich Zusatzwerkzeuge, wie z.B. die Google-Toolbar heraus, welche die Ausführung von einigen Javascript-Funktionen blockierten und sich zusätzlich negativ auf die Performance der Webanwendung *TRIP* Highway auswirkten. Das Ausblenden dieser Zusatzwerkzeuge im jeweiligen Browserfenster hat diese Schwierigkeiten gelöst.

4.1.3 Test und Qualitätssicherung

Während der Implementation habe ich die Webanwendung dauerhaft mit dem Internet Explorer 7 getestet und Fehlerfälle somit sofort behoben.

Bei wichtigen Entwicklungsschritten wurde die Anwendung zusätzlich auf einem, für einen Redakteur konfigurierten, Testrechner mit dem Internet Explorer 6 getestet, da dieser Browser zur Zeit noch auf den meisten PCs im Hause RTL installiert ist.

Der Test mit anderen Webbrowsern wie Opera, Mozilla Firefox oder Netscape entfiel, da diese weder bei der Standardauslieferung von neuen Computern installiert werden, noch von der *EDV* unterstützt und gewartet werden.

Die Auftraggeber der Firma RTL Creation testeten die Endversion. Entdeckte kleinere Mängel wurden anschließend behoben und schließlich wurde die verbesserte Version erfolgreich freigegeben.

4.2 Bildvergleich der Vorschaubilder

Wie schon im oberen Kapitel angesprochen, ist der Bildvergleich die sinnvollste Methode um einen Datensatz im Creationarchiv eindeutig zu identifizieren und diesen danach entsprechend hervorzuheben, damit dem Redakteur die Suche erleichtert wird.

Da das Previewbild selten zwischen dem Newsarchiv und dem Creationarchiv aufgrund von unterschiedlichen Arbeitsweisen der Mediendokumentare übereinstimmt, muss überprüft werden, ob sich die Integration eines solchen vergleichenden Programms in die webbasierte Anwendung *TRIPHighway* lohnt.

Im Rahmen der Arbeit wurde von mir zunächst ein geeignetes Bildvergleichsverfahren ausgewählt. Auf der Entscheidung basierend habe ich vier mathematisch verschiedene Ansätze implementiert und anhand von Realbildern mit Hilfe des java-basierten Programms ImageJ³⁷ getestet. Abschließend wurden diese Testergebnisse ausgewertet.

³⁷ weitere Informationen im Kapitel 6.2.4 „ImageJ“

4.2.1 Bildvergleich-Verfahren

Um ein geeignetes Verfahren zu bestimmen, mit dem man die Bilder vergleichen kann, muss man zuerst wissen, in welcher Weise die Bilder übereinstimmen und Ähnlichkeiten bzw. Parallelen aufweisen.

Die Erstellung der Bilder stellt in diesem Fall einen wichtigen Faktor dar. Alle im Archivsystem abgespeicherten und mit Hilfe des *JPEG*³⁸-Verfahrens komprimierten Bilder sind einzelne Frames, die aus den dazugehörigen Videos mit Hilfe der Software *VPMS PreviewClient* exportiert wurden und den Inhalt des Videos durch diese Momentaufnahme charakterisieren sollen.

Diese Informationen sind in dem Zusammenhang wesentlich, da die Erstellung der Previewbilder mit der gleichen Software – trotz unterschiedlicher Archivsysteme – eine gute Basis bildet. Damit kann festgelegt werden, dass sich die exportierten Frames in der Bildgröße, Farbtiefe und dem Kompressionsverfahren nicht unterscheiden.

Trotzdem muss man berücksichtigen, dass die Farbwerte zwischen den beiden zu vergleichenden Bildern in manchen Fällen nicht übereinstimmen oder dass die Bilder schwarze, schmale Balken an den Rändern beinhalten. Der Grund dafür liegt in der unterschiedlichen Aufzeichnung und Codierung des Videomaterials. Die Aufzeichnung erfolgt über das *VPMS/INES*³⁹-System und die Transcodierung des Vorschaumaterials durch das *VPMS* Werkzeug *Balance TV*.

Ein Lösungsansatz wäre der direkte Bildvergleich mittels „*Template Matching*“. Als *Template* bezeichnet man in diesem Fall ein Teilbild innerhalb eines größeren Bildes. So versucht man das gesuchte Bildmuster (*Template*) in einem anderen Bild wiederzufinden, indem man das *Template* über das Bild bewegt und die Differenz gegenüber dem darunterliegenden Teilbild misst.⁴⁰

Das Problem ist, dass die Archivierung in unterschiedlichen Systemen von verschiedenen Mediendokumentaren durchgeführt wird und deshalb die Wahrscheinlichkeit sehr gering ist, dass zwei Archivare den gleichen Frame eines Beitrags exportieren und somit als Vorschaubild festlegen (siehe Abbildung 4-10).

³⁸ Joint Photographic Experts Group

³⁹ Integrated News Editing System

⁴⁰ vgl. [BurBur 2006], S.412 - 413

Leider gibt es momentan in diesem Umfeld keine Regelungen oder Standards zwischen den Archiven, da zuvor nicht archivübergreifend gesucht wurde.

Der Ansatz des direkten Bildvergleichs mittels „Template Matching“ ist aus diesem Grund unbrauchbar.



Abbildung 4-10: Testbilder Bildvergleich

Ein weiterer Lösungsansatz ist der Vergleich der „Histogramme von Farbbildern“.

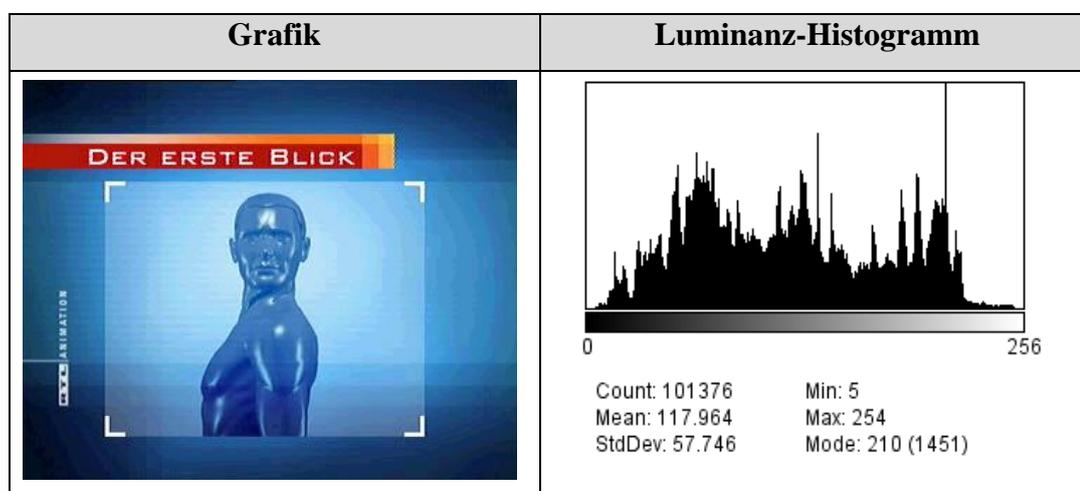
Histogramme sind Häufigkeitsverteilungen und Histogramme von Bildern beschreiben die Häufigkeit der einzelnen Intensitätswerte. Mit Histogrammen von Farbbil-

dem beschreibt man entweder die Häufigkeitsverteilung der Helligkeits⁴¹-Werte, oder die Histogramme der einzelnen Farbkanäle.⁴²

Dieser Ansatz könnte im Gegensatz zum Bildvergleich mittels „Template Matching“ durchaus positive Ergebnisse erzielen, da er bei den Farb- oder Helligkeitswerten der einzelnen Pixel ansetzt, ohne die räumliche Verteilung der Pixel im Bild zu beachten. Daher können unterschiedliche Frames, die zeitnah aus dem Video exportiert wurden und aus derselben Sequenz stammen, durchaus ähnliche Histogramme haben.

4.2.2 Histogramm-Analyse

Damit man sich einen Überblick verschaffen kann, ob Luminanz-Histogramme oder die Histogramme der einzelnen Farbkanäle besser für den Bildvergleich verwendet werden können, wurden einzelne Grafiken des Creationarchivs mit Hilfe der Software ImageJ zu Testzwecken gegenübergestellt. Hier folgt ein Beispiel, welches stellvertretend für alle Vergleiche der Luminanz-Histogramme zwischen News- und Creationarchiv steht:



⁴¹ sogenannte Luminanz

⁴² vgl. [BurBur 2006], S.39, 49

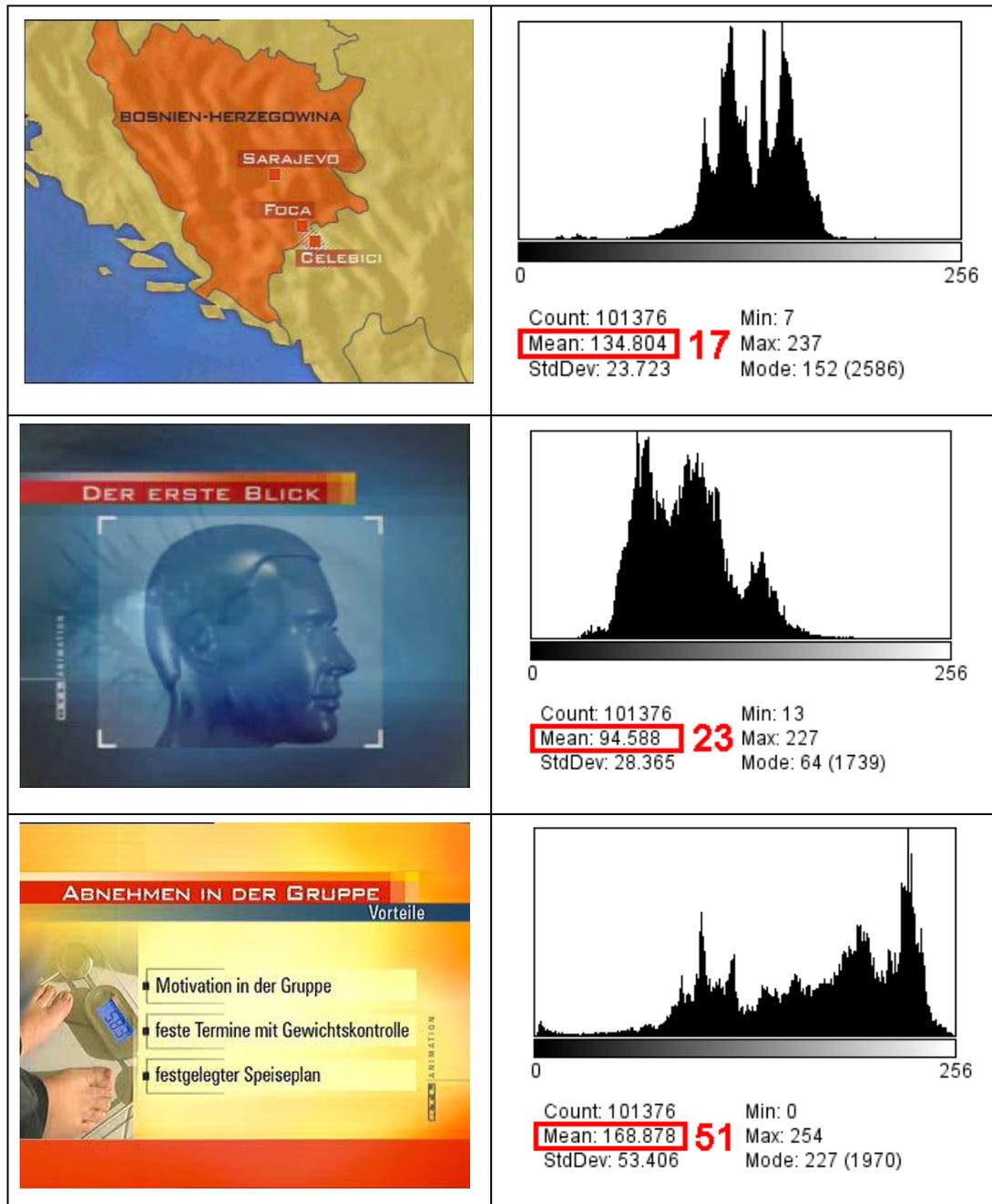


Abbildung 4-11: Vergleich der Luminanz-Histogramme

In der Abbildung 4-11 sieht man auf der linken Seite Beispielvergleichsgrafiken und auf der rechten Seite deren Luminanz-Histogramme. Die zu vergleichenden Mean⁴³-Werte sind rot eingerahmt. Neben dem Rahmen befindet sich jeweils die Häufigkeitsdifferenz der Grafik (in rot markiert), die den Unterschied im Vergleich zur ersten Grafik mit der Überschrift „Der erste Blick“ kennzeichnen soll.

⁴³ engl. Häufigkeit

Das Ergebnis dieses Beispiels zeigt, dass die Differenz der Häufigkeiten zwischen den ersten beiden Bildern, welche aus verschiedenen Sequenzen stammen, geringer ist, als zwischen dem ersten und dritten Bild, die beide aus einer Sequenz exportiert wurden. Bei weiteren analog aufgebauten Testfällen wurden die Bildpaare⁴⁴ ebenfalls nicht richtig zugeordnet.

Zusammenfassend lässt sich daraus schließen, dass der Vergleich der Mittelwerte der Luminanz-Histogramme zu ungenau ist und man keine eindeutige Aussage darüber treffen kann, welche Bildpaare sich inhaltlich ähneln.

Ein weiterer Ansatz in Bezug auf die Luminanz-Histogramme wäre der Vergleich des gesamten Histogrammverlaufs mit seinen Minima und Maxima. Allerdings werden bei diesem Vergleich diverse Schwierigkeiten auftreten, da der Histogrammverlauf durch unterschiedliche Globalhelligkeit, andersartigen Globalkontrast oder ungleiche Farbsättigung unterschiedlich gestaucht und verschoben sein kann.

Aus diesen Gründen wird untersucht, ob ein Vergleich der Histogramme der einzelnen Farbkanäle ein besseres Ergebnis erzielt. Die Gegenüberstellung der Farbkanäle liefert verschiedene Ausgangspunkte; vier Ansätze habe ich ausgewählt und jeweils mit Hilfe eines Zusatzprogramms (sogenanntes Plugin) implementiert und mit ImageJ getestet.

Vergleich der Mittelwerte. Dieser Ansatz wurde ausgewählt, da er isolierte Spitzen und Löcher⁴⁵ in den Farbhistogrammen nicht berücksichtigt und größere, unerwünschte Farbänderungen zwischen den Vorschaubildern auf das Ergebnis der Histogrammanalyse wenig Einfluss nehmen.

Das entwickelte Plugin liest ein Bild ein und speichert für jeden Pixel die einzelnen Farbwerte (Rot, Grün, Blau) in einem zweidimensionalen Datenfeld (Array). Daraufhin werden alle einzelnen Werte des aktuell gesuchten Bildes (U) und alle Werte des zu vergleichenden Bildes (V) voneinander getrennt, über die Menge der Intervalle (sogenannte Bins), summiert (siehe Abbildung 4-12). Die Differenz der Summen von U und V zum Betrag, bilden die einzelnen Farbkanaldifferenzen (Diff_{Rot} , $\text{Diff}_{\text{Grün}}$, $\text{Diff}_{\text{Blau}}$).

⁴⁴ Ein Bildpaar besteht aus dem Sequenz-Vorschaubild aus dem Newsarchiv und dem dazu passenden Vorschaubild aus dem Creationarchiv

⁴⁵ weitere Informationen im Anhang Kapitel 6.1.5 „Spitzen und Löcher“

Die Gesamtdifferenz der Mittelwerte ($\text{Diff}_{\text{Gesamt}}$) ergibt sich aus der Addition der einzelnen Farbkanaldifferenzen und der Division durch die Pixelanzahl des Bildes:

$$\begin{aligned}\text{Diff}_{\text{Rot}} &= \left| \sum_{i=0}^{255} V_R(i) - \sum_{i=0}^{255} U_R(i) \right| \\ \text{Diff}_{\text{Gruen}} &= \left| \sum_{i=0}^{255} V_G(i) - \sum_{i=0}^{255} U_G(i) \right| \\ \text{Diff}_{\text{Blau}} &= \left| \sum_{i=0}^{255} V_B(i) - \sum_{i=0}^{255} U_B(i) \right| \\ \text{Diff}_{\text{Gesamt}} &= \frac{\text{Diff}_{\text{Rot}} + \text{Diff}_{\text{Gruen}} + \text{Diff}_{\text{Blau}}}{\text{AnzahlPixel}}\end{aligned}$$

Abbildung 4-12: Berechnung der Mittelwert-Differenzen

Um die Abweichungen zwischen den einzelnen Differenzen vom Suchbild besser auszuwerten, wurde zunächst ein prozentuales Vergleichsmaß mit Hilfe dieser Formel berechnet:

$$\text{Distanz} = \frac{x * 100}{x_1}$$

Abbildung 4-13: Formel - prozentuales Vergleichsmaß

Der Wert „Distanz“ lässt sich durch die klassische Prozentrechnung ermitteln. Hierbei wird die Gesamtdifferenz des aktuellen Bildes (x) mit 100 multipliziert und durch die niedrigste Gesamtdifferenz aus der Menge der paarweisen Vergleiche zum aktuellen Suchbild (x_1) dividiert. Die Distanz enthält nur für das Bild, welches dem gesuchten Bild am meisten ähnelt, den Wert 100. Die restlichen Bilder weisen Distanzwerte größer 100 auf. Durch die Abstandsbestimmung zweier Werte (z.B. $132 - 100 = 32$) lässt sich eine Aussage darüber treffen, in wie weit sich die verglichenen Bilder ähneln: Je kleiner der Abstand, desto mehr Ähnlichkeit.

Der Nachteil dieser Formel ist, dass die Variable x_1 den Wert 0 annehmen und der Distanzwert in diesem Fall nicht bestimmt werden kann. Dieser Umstand tritt dann auf, wenn ein Bild aus der Vergleichsmenge identisch zum gesuchten Bild ist und die

Differenz somit 0 aufweist. Da es in solchen Fällen nicht möglich ist, die Distanz mit Hilfe der Formel zu berechnen, haben wir eine andere Formel für ein Vergleichsmaß entwickelt, welche in jedem Fall einen Wert berechnen kann:

$$\text{Distanz} = \frac{M - x}{M - x_1}$$

Abbildung 4-14: Formel - Vergleichsmaß

Hierbei wird die Gesamtdifferenz des aktuellen Bildes (x) vom Mittelwert der Gesamtdifferenzen aller paarweisen Vergleiche (M) subtrahiert. Ebenso wird die niedrigste Gesamtdifferenz aus der Menge der paarweisen Vergleiche zum aktuellen Suchbild (x_1) vom Mittelwert (M) subtrahiert. Diese beiden Teilergebnisse werden dividiert und ergeben ein Vergleichsmaß, welches ebenfalls Rückschlüsse über den Grad der Abweichung zum gesuchten Bild zulässt.

Im besten Fall sind die Bilder komplett identisch, dann ist $x = x_1$ und die Distanz enthält den Wert 1. Liegt das Ergebnis zwischen 0 und 1, ähneln sich die Bilder in einem gewissen Grad. Fällt der Wert unter 0, ist die Differenz der eben verglichenen Bilder so stark, dass kaum bis gar keine Ähnlichkeiten vorhanden sind.

Vorteile gegenüber der Formel für das prozentuale Vergleichsmaß (Abbildung 4-13) sind, dass das Vergleichsmaß auch für $x = x_1 = 0$ Werte enthält und für kleine x Werte keine enorm großen Werte liefert.

Die folgende Abbildung 4-15 enthält die Distanzerggebnisse für den „Vergleich der Mittelwerte“:

	abnehmen_original_creation.jpg	abnehmen_original_news.jpg	arbeitslose_creation.jpg	arbeitslose_news.jpg	ersterBlick_original_creation.jpg	ersterBlick_original_news.jpg	magnetfeld_creation.jpg	magnetfeld_news.jpg	soldaten_original_creation.jpg	soldaten_original_news.jpg	sonnenspiegel_creation.jpg	sonnenspiegel_news.jpg	stufenmodell_creation.jpg	stufenmodell_news.jpg	vogelgrippe_creation.jpg	vogelgrippe_news.jpg
abnehmen_original_creation.jpg	-	X	-0,45	-0,25	-0,73	-0,61	-0,92	-1,16	0,82	0,53	-0,13	-0,13	-0,41	-0,27	0,89	0,52
abnehmen_original_news.jpg	X	-	-0,57	-0,33	-0,89	-0,76	-0,86	-1,14	0,93	0,75	-0,20	-0,20	-0,52	-0,36	0,98	0,80
arbeitslose_creation.jpg	-1,17	-1,05	-	0,74	0,70	0,15	-0,51	-0,81	-0,58	-0,39	0,65	0,65	X	0,64	-0,64	-0,44
arbeitslose_news.jpg	-1,45	-1,27	0,95	-	0,51	0,13	-0,83	-1,27	-0,59	-0,31	0,99	0,99	X	0,83	-0,68	-0,39
ersterBlick_original_creation.jpg	-1,48	-1,34	X	0,71	-	0,63	-0,10	-0,44	-0,82	-0,61	0,55	0,55	0,94	0,59	-0,89	-0,67
ersterBlick_original_news.jpg	-1,80	-1,62	0,40	0,44	0,72	-	0,64	0,31	-0,95	-0,67	0,61	0,61	0,35	X	-1,03	-0,75
magnetfeld_creation.jpg	-1,05	-0,73	-0,03	-0,01	0,15	0,65	-	X	-0,55	-0,24	0,09	0,09	-0,06	0,31	-0,64	-0,24
magnetfeld_news.jpg	-0,90	-0,63	-0,05	-0,03	0,10	0,56	X	-	-0,48	-0,23	0,05	0,05	-0,08	0,23	-0,55	-0,22
soldaten_original_creation.jpg	0,61	0,77	-0,47	-0,19	-0,84	-0,69	-1,07	-1,39	-	0,88	-0,04	-0,04	-0,42	-0,23	X	0,86
soldaten_original_news.jpg	0,09	0,41	-0,36	-0,10	-0,70	-0,56	-0,73	-1,03	0,69	-	0,04	0,04	-0,31	-0,13	0,58	X
sonnenspiegel_creation.jpg	-1,07	-0,92	0,54	0,68	0,15	0,14	-0,60	-0,95	-0,39	-0,17	-	X	0,60	0,70	-0,46	-0,24
sonnenspiegel_news.jpg	-1,07	-0,92	0,54	0,68	0,15	0,14	-0,60	-0,95	-0,39	-0,17	X	-	0,60	0,70	-0,46	0,60
stufenmodell_creation.jpg	-1,15	-1,03	X	0,77	0,65	0,10	-0,57	-0,87	-0,55	-0,36	0,69	0,69	-	0,59	-0,61	-0,41
stufenmodell_news.jpg	-1,63	-1,44	0,78	0,81	0,33	0,67	-0,33	-0,80	-0,73	-0,44	X	X	0,72	-	-0,82	-0,52
vogelgrippe_creation.jpg	0,71	0,84	-0,46	-0,19	-0,81	-0,67	-1,08	-1,39	X	0,77	-0,05	-0,05	-0,41	-0,23	-	0,76
vogelgrippe_news.jpg	0,10	0,47	-0,38	-0,13	-0,71	-0,57	-0,68	-0,97	0,69	X	0,01	0,01	-0,33	-0,16	0,58	-

Abbildung 4-15: Ergebnis des "Vergleich der Mittelwerte"

In der ersten Spalte und in der ersten Zeile der Tabelle sind die Namen der Testbilder aufgelistet, darunter folgen die Distanzerggebnisse in Form einer Matrix.

Für den Test wurden 16 Vorschaubilder ausgewählt, da diese Anzahl genau der durchschnittlichen Bildanzahl beim realen Vergleich entspricht.

Ebenso wurde bei der Bildauswahl darauf geachtet, dass möglichst alle unterschiedlichen Arten von Realfällen (siehe z.B. Abbildung 4-10) enthalten sind:

- Fall 1: Die Vorschaubilder sind nahezu identisch
- Fall 2: Die Vorschaubilder zeigen unterschiedliche Einstellungen der animierten Grafiken (z.B. Deutschlandkarte Übersicht, Deutschlandkarte NRW)
- Fall 3: Ein Vorschaubild wurde in einer Überblendung zwischen zwei Sequenzen erstellt und enthält Bildteile und Farbwerte beider Sequenzen

- Fall 4: Die Vorschaubilder sind semantisch sehr ähnlich, aber eins der Bilder beinhaltet schwarze, vertikale Streifen am Bildrand

Die Tabelle in Abbildung 4-15 wird zeilenweise gelesen; dabei steht in der ersten Spalte der Dateiname des Bildes, welches mit den 15 anderen Bildern verglichen wird.

Das Bild mit der geringsten Distanz wird mit einem roten Kreuz in der Tabelle gekennzeichnet; das Vorschaubild mit der zweitniedrigsten Distanz (kleiner 1) wird in blauer Farbe hervorgehoben und kennzeichnet damit den Zweitplatzierten. Alle Werte größer 0 sind fett formatiert, während alle Werte kleiner 0 kursiv formatiert und in einer kleineren Schriftgröße dargestellt sind.

Im Vergleich der Testbilder werden 6 von 16 Testbilder ihrem Gegenbild aus dem Creationarchiv korrekt zugeordnet. Dies entspricht einer Trefferquote von ungefähr 37,5 %.

Bei der Auswertung des Ergebnisses fiel auf, dass nur Bildpaare korrekt zugeordnet werden konnten, die sich sehr ähnlich waren:

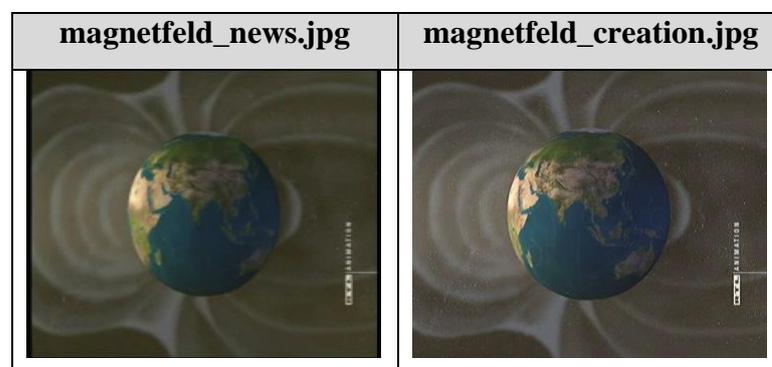


Abbildung 4-16: Beispiel für korrekte Zuordnung beim Mittelwertvergleich

Der Vergleich der Mittelwerte ist also für die verschiedenen Arten von auftretenden Realfällen eher ungeeignet, da er nur eine grobe Zuordnung zulässt. Um eine feinere Einordnung zu erlangen und damit eventuell die Trefferquote zu erhöhen, implementierte man den nächsten Ansatz.

Vergleich der Absolutwerte. Anders als im ersten Ansatz haben hier isolierte Spitzen und Löcher in Histogrammen mehr Einfluss auf das Ergebnis des Vergleichs. Der Vorteil ist, dass Bilder mit einem großen Spektrum an Farbwerten oder mit hohen Intensitätswerten in den äußeren Farbbereichen eher dem richtigen Gegenbild

zugeordnet werden, als bei dem Vergleich der Mittelwerte. Grundlage für die Bildung der Differenzen waren diese Formeln:

$$\text{Diff}_{Rot} = \sum_{i=0}^{255} |V_R(i) - U_R(i)|$$

$$\text{Diff}_{Gruen} = \sum_{i=0}^{255} |V_G(i) - U_G(i)|$$

$$\text{Diff}_{Blau} = \sum_{i=0}^{255} |V_B(i) - U_B(i)|$$

$$\text{Diff}_{Gesamt} = \text{Diff}_{Rot} + \text{Diff}_{Gruen} + \text{Diff}_{Blau}$$

Abbildung 4-17: Berechnung der Absolutwert-Differenzen

Bei diesem mathematischen Ansatz wird die Differenz zwischen jedem einzelnen Intensitätswert des aktuellen Vergleichsbildes (V) und dem Intensitätswert des Suchbildes (U) zum Betrag gebildet und über die Anzahl Bins summiert. Daraus ergeben sich dann die Differenzen der einzelnen Farbkanäle (Diff_{Rot} , Diff_{Gruen} , Diff_{Blau}). Durch Addition der drei Differenzwerte ergibt sich die Summe der Absolutwerte (Diff_{Gesamt}).

Nach Berechnung des Vergleichmaßes (Abbildung 4-14) werden die Testbilder folgendermaßen eingeordnet:

	abnehmen_original_creation.jpg	abnehmen_original_news.jpg	arbeitslose_creation.jpg	arbeitslose_news.jpg	ersterBlick_original_creation.jpg	ersterBlick_original_news.jpg	magnetfeld_creation.jpg	magnetfeld_news.jpg	soldaten_original_creation.jpg	soldaten_original_news.jpg	sonnenspiegel_creation.jpg	sonnenspiegel_news.jpg	stufenmodell_creation.jpg	stufenmodell_news.jpg	vogelgrippe_creation.jpg	vogelgrippe_news.jpg
abnehmen_original_creation.jpg	-	X	-0,04	0,26	0,09	-0,40	-1,21	-1,21	-0,39	-0,41	-0,74	-0,74	0,14	0,15	0,64	-0,05
abnehmen_original_news.jpg	X	-	-0,19	0,27	-0,05	-0,29	-1,29	-1,32	-0,22	-0,30	-0,74	-0,74	-0,04	0,23	0,48	0,13
arbeitslose_creation.jpg	-0,22	-0,26	-	0,50	0,68	-0,14	-0,58	-0,59	-0,37	-0,51	-0,38	-0,38	X	0,15	-0,12	-0,24
arbeitslose_news.jpg	-0,08	-0,02	0,85	-	0,97	-0,50	-1,08	-1,07	-0,38	-0,66	-0,56	-0,56	X	0,22	-0,25	-0,35
ersterBlick_original_creation.jpg	-0,29	-0,35	X	0,83	-	-0,01	-0,80	-0,82	-0,64	-0,76	-0,53	-0,53	0,99	0,41	-0,24	-0,45
ersterBlick_original_news.jpg	-0,68	-0,48	-0,18	-0,46	0,13	-	0,19	-0,04	-0,67	-0,55	0,19	0,19	-0,23	X	-0,59	-0,50
magnetfeld_creation.jpg	-0,68	-0,63	-0,33	-0,40	-0,21	0,40	-	X	-0,51	-0,40	0,52	0,52	-0,38	0,16	-0,56	-0,49
magnetfeld_news.jpg	-0,56	-0,53	-0,25	-0,30	-0,14	0,31	X	-	-0,59	-0,44	0,32	0,32	-0,28	0,09	-0,47	-0,40
soldaten_original_creation.jpg	-0,66	-0,20	-0,50	-0,05	-0,52	-0,65	-1,68	-2,25	-	X	-0,22	-0,22	-0,44	-0,22	0,37	0,90
soldaten_original_news.jpg	-0,49	-0,21	-0,68	-0,40	-0,52	-0,29	-0,97	-1,28	0,76	-	-0,40	-0,40	-0,68	0,32	0,31	X
sonnenspiegel_creation.jpg	-0,35	-0,31	-0,20	-0,17	-0,14	0,09	0,16	0,01	-0,13	-0,21	-	X	-0,21	0,10	-0,36	-0,28
sonnenspiegel_news.jpg	-0,35	-0,31	-0,20	-0,17	-0,14	0,09	0,16	0,01	-0,13	-0,21	X	-	-0,21	0,10	-0,36	-0,28
stufenmodell_creation.jpg	-0,14	-0,20	X	0,25	0,67	-0,19	-0,65	-0,64	-0,37	-0,54	-0,41	-0,41	-	0,13	-0,10	-0,19
stufenmodell_news.jpg	-0,53	-0,35	0,11	0,05	0,49	X	-0,64	-0,92	-0,95	-0,48	-0,09	-0,09	0,09	-	-0,59	-0,40
vogelgrippe_creation.jpg	X	0,78	0,07	-0,06	0,07	-0,80	-2,16	-2,21	0,13	0,15	-1,67	-1,67	0,20	0,02	-	0,82
vogelgrippe_news.jpg	-0,26	0,15	-0,29	-0,22	-0,33	-0,54	-1,71	-1,76	0,61	X	-1,10	-1,10	-0,11	0,24	0,72	-

Abbildung 4-18: Ergebnis des "Vergleich der Absolutwerte"

Die Tabelle liest sich analog zum „Vergleich der Mittelwerte“.

In diesem Test werden 7 von 16 Testbildern ihrem Gegenbild korrekt zugeordnet, was einer Trefferquote von 43,75 % entspricht. Dabei werden die gleichen drei Bildpaare wie im ersten Ansatz korrekt zugeordnet. Zusätzlich wird dieses Bildpaar einseitig (Creationarchiv → Newsarchiv) erkannt:

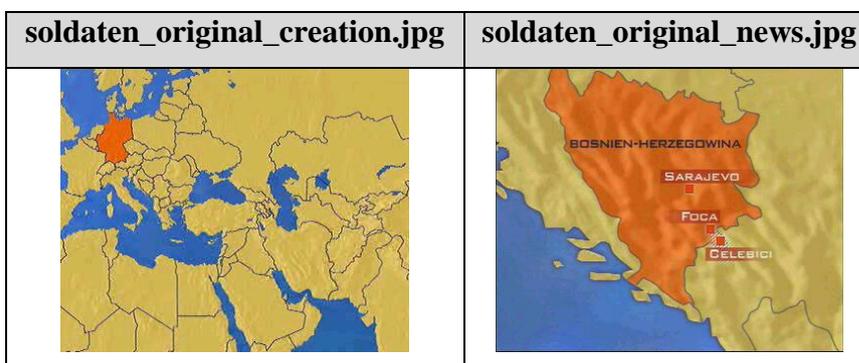


Abbildung 4-19: Beispiel für korrekte Zuordnung beim Absolutwertvergleich

Den entschiedenen Ausschlag für die Erkennung geben die isolierten Spitzen in den Farbhistogrammen des Creationarchiv-Bildes.

Genau wie im ersten Ansatz „Vergleich der Mittelwerte“ werden auch hier nur die Intensitätswerte der *einzelnen* Farbkomponenten miteinander verglichen.

Diese beiden Ansätze können jedoch keine Informationen über die Verteilung der tatsächlichen Farben in einem Bild geben, da das räumliche Zusammentreffen der Farbkomponenten innerhalb eines Bildelements nicht berücksichtigt wird.⁴⁶ Deswegen wird der dritte Ansatz verfolgt.

Vergleich der 2D-Farbhistogramme. Da ein volles Histogramm des RGB-Farbraums dreidimensional wäre und 2^{24} Werte für ein Bild abgespeichert werden müssten, werden nicht alle drei Farbkanäle in einem Histogramm kombiniert. Sinnvoller ist es, zweidimensionale Projektionen darzustellen. In diesem Fall wurden die drei Projektionen (Rot→Grün, Rot→Blau und Grün→Blau) gewählt. Somit ergeben sich für die Differenzen folgende Formeln:

$$\text{Diff}_{\text{RotGruen}} = \sum_{i=0}^{255} \sum_{j=0}^{255} |V_{RG}(i, j) - U_{RG}(i, j)|$$

$$\text{Diff}_{\text{RotBlau}} = \sum_{i=0}^{255} \sum_{j=0}^{255} |V_{RB}(i, j) - U_{RB}(i, j)|$$

$$\text{Diff}_{\text{GruenBlau}} = \sum_{i=0}^{255} \sum_{j=0}^{255} |V_{GB}(i, j) - U_{GB}(i, j)|$$

$$\text{Diff}_{\text{Gesamt}} = \text{Diff}_{\text{RotGruen}} + \text{Diff}_{\text{RotBlau}} + \text{Diff}_{\text{GruenBlau}}$$

Abbildung 4-20: Berechnung der Projektion-Differenzen

Bei dieser Berechnung werden zwei ineinander verschachtelte Summen berechnet und das Ergebnis der einzelnen Farbkomponenten in einer Gesamtsumme ($\text{Diff}_{\text{Gesamt}}$) addiert. Nach der Ausführung des Plugins in ImageJ gelangt man zu folgendem Ergebnis der Distanzwerte:

⁴⁶ vgl. [BurBur 2006], S.50

	abnehmen_original_creation.jpg	abnehmen_original_news.jpg	arbeitslose_creation.jpg	arbeitslose_news.jpg	ersterBlick_original_creation.jpg	ersterBlick_original_news.jpg	magnetfeld_creation.jpg	magnetfeld_news.jpg	soldaten_original_creation.jpg	soldaten_original_news.jpg	sonnenspiegel_creation.jpg	sonnenspiegel_news.jpg	stufenmodell_creation.jpg	stufenmodell_news.jpg	vogelgrippe_creation.jpg	vogelgrippe_news.jpg
abnehmen_original_creation.jpg	-	X	-0,91	-0,82	-1,13	-1,06	-1,00	-1,01	-0,24	-0,59	-1,08	-1,08	-0,89	-0,65	0,51	-0,20
abnehmen_original_news.jpg	X	-	-1,11	-0,75	-1,32	-1,19	-1,15	-1,08	-0,20	-0,43	-1,16	-1,16	-1,12	-0,60	0,48	-0,07
arbeitslose_creation.jpg	-0,33	-0,34	-	0,20	0,53	-0,29	-0,43	-0,43	-0,26	-0,34	-0,43	-0,43	X	-0,27	-0,16	-0,17
arbeitslose_news.jpg	-0,77	-0,62	0,67	-	X	-0,59	-1,08	-1,09	-0,69	-0,86	-1,04	-1,04	0,74	-0,37	-0,29	-0,21
ersterBlick_original_creation.jpg	-0,66	-0,66	0,95	0,57	-	-0,43	-0,75	-0,75	-0,40	-0,51	-0,72	-0,72	X	-0,34	-0,18	-0,20
ersterBlick_original_news.jpg	-0,45	-0,43	-0,33	-0,25	-0,27	-	-0,31	-0,41	-0,31	-0,35	-0,28	-0,28	-0,33	X	-0,36	-0,29
magnetfeld_creation.jpg	-0,42	-0,41	-0,56	-0,54	-0,58	-0,28	-	X	-0,50	-0,54	0,15	0,15	-0,57	-0,26	-0,51	-0,47
magnetfeld_news.jpg	-0,37	-0,32	-0,50	-0,48	-0,51	-0,34	X	-	-0,46	-0,48	-0,04	-0,04	-0,51	-0,28	-0,47	-0,41
soldaten_original_creation.jpg	-0,50	-0,33	-0,96	-1,06	-0,86	-1,06	-1,61	-1,68	-	0,64	-1,59	-1,59	-0,92	-0,78	0,50	X
soldaten_original_news.jpg	-0,84	-0,44	-1,31	-1,34	-1,13	-1,14	-1,78	-1,79	X	-	-1,65	-1,65	-1,38	-0,58	0,02	0,97
sonnenspiegel_creation.jpg	-0,16	-0,15	-0,19	-0,18	-0,19	-0,11	-0,02	-0,07	-0,17	-0,17	-	X	-0,18	-0,06	-0,18	-0,16
sonnenspiegel_news.jpg	-0,16	-0,15	-0,19	-0,18	-0,19	-0,11	-0,02	-0,07	-0,17	-0,17	X	-	-0,18	-0,06	-0,18	-0,16
stufenmodell_creation.jpg	-0,34	-0,35	X	0,21	0,56	-0,31	-0,45	-0,45	-0,27	-0,36	-0,43	-0,43	-	-0,26	-0,13	-0,16
stufenmodell_news.jpg	-0,44	-0,36	-0,45	-0,26	-0,32	X	-0,44	-0,52	-0,36	-0,33	-0,22	-0,22	-0,39	-	-0,39	-0,26
vogelgrippe_creation.jpg	0,22	0,23	-0,66	-0,57	-0,42	-1,41	-1,85	-1,91	0,31	-0,38	-1,83	-1,86	-0,47	-1,05	-	X
vogelgrippe_news.jpg	-0,75	-0,46	-0,82	-0,51	-0,57	-1,35	-1,93	-1,90	0,80	0,38	-1,83	-1,83	-0,71	-0,82	X	-

Abbildung 4-21: Ergebnis des "Vergleich der 2D-Farbhistogramme"

Der Bildvergleich mit Hilfe der drei Projektionen (RG, RB, GB) erreicht das beste Testergebnis. Hier erhält man eine Trefferquote von 56,25 %, da 9 von 16 Testbildern richtig zugewiesen werden. Das Ergebnis wird zusätzlich noch durch den größten Abstand zwischen dem ähnlichsten und dem zweitähnlichsten Bild (blau markiert) aufgewertet.

Zusätzlich zu den schon vorher korrekt erkannten Bildpaaren, ordnet dieser Ansatz Bilder einander zu, die sich auf den ersten Blick semantisch sehr ähneln. Bei genauem Betrachten entdeckt man aber Videocodierungsfehler in Form von schwarzen, vertikalen Streifen (hier in der Grafik: „vogelgrippe_news.jpg“), die den Bildvergleich beeinflussen:

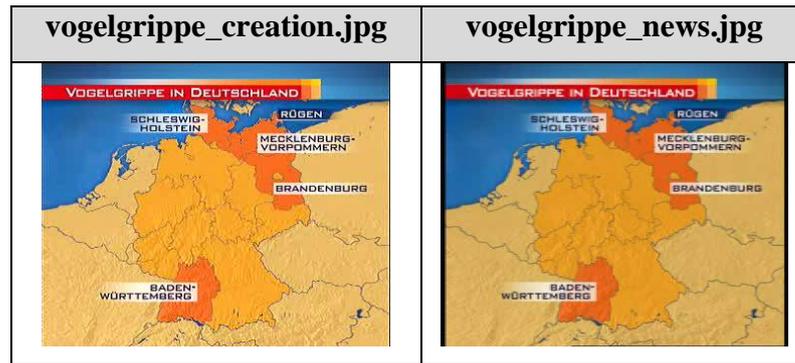


Abbildung 4-22: Beispiel für korrekte Zuordnung beim 2D-Histogrammvergleich

Die Trefferquote der obigen drei Bildvergleich-Methoden steigert sich kontinuierlich, ist aber noch nicht zufriedenstellend. Deswegen folgt ein Ansatz, der durch eine Kombination zweier Vorgehensweisen versucht, deren Nachteile zu kompensieren.

Kombination der Farbhistogramm-Vergleiche. Diese Methode vereint die Ansätze „Vergleich der Absolutwerte“ und „Vergleich der 2D-Farbhistogramme“ mit dem Ziel, die besten Trefferquoten zu liefern.

Auf der einen Seite werden die absoluten Differenzen der einzelnen Farbkanäle berücksichtigt; auf der anderen Seite werden die 2D-Histogramm Differenzen mit einbezogen, die Rückschlüsse auf das Auftreten zweier Farben in einem Pixel zulassen. Da die absoluten Differenzen (Diff_{Rot} , $\text{Diff}_{\text{Grün}}$, $\text{Diff}_{\text{Blau}}$) nur über 256 Bins summiert werden, werden sie jeweils mit dem Faktor 256 gewichtet. Auf diese Weise nehmen sie einen gleich großen Einfluss auf die Gesamtdifferenz, wie die 2D-Histogramm Differenzen, die über $256 * 256$ Bins summiert werden.

Grundlage für den kombinierten Vergleich ist demnach folgende Formel:

$$\text{Diff}_{\text{Gesamt}} = \text{Diff}_{\text{RotGrün}} + \text{Diff}_{\text{RotBlau}} + \text{Diff}_{\text{GrünBlau}} + (256 * \text{Diff}_{\text{Rot}}) + (256 * \text{Diff}_{\text{Grün}}) + (256 * \text{Diff}_{\text{Blau}})$$

Abbildung 4-23: Kombination der Differenzen

Mit Hilfe der Berechnung der Distanzwerte (Abbildung 4-14) können die Vergleichsmaße übersichtlich in folgender Tabelle präsentiert werden:

	abnehmen_original_creation.jpg	abnehmen_original_news.jpg	arbeitslose_creation.jpg	arbeitslose_news.jpg	ersterBlick_original_creation.jpg	ersterBlick_original_news.jpg	magnetfeld_creation.jpg	magnetfeld_news.jpg	soldaten_original_creation.jpg	soldaten_original_news.jpg	sonnenspiegel_creation.jpg	sonnenspiegel_news.jpg	stufenmodell_creation.jpg	stufenmodell_news.jpg	vogelgrippe_creation.jpg	vogelgrippe_news.jpg
abnehmen_original_creation.jpg	-	X	-0,04	0,26	0,08	-0,40	-1,21	-1,21	-0,39	-0,41	-0,74	-0,74	0,14	0,15	0,64	-0,05
abnehmen_original_news.jpg	X	-	-0,19	0,27	-0,06	-0,29	-1,29	-1,32	-0,22	-0,30	-0,74	-0,74	-0,04	0,23	0,48	0,13
arbeitslose_creation.jpg	-0,22	-0,26	-	0,50	0,68	-0,14	-0,58	-0,59	-0,37	-0,51	-0,38	-0,38	X	0,15	-0,12	-0,23
arbeitslose_news.jpg	-0,08	-0,02	0,85	-	0,97	-0,50	-1,08	-1,07	-0,38	-0,66	-0,56	-0,56	X	0,22	-0,25	-0,35
ersterBlick_original_creation.jpg	-0,29	-0,35	X	0,83	-	-0,01	-0,80	-0,82	-0,64	-0,76	-0,53	-0,53	0,99	0,41	-0,24	-0,45
ersterBlick_original_news.jpg	-0,68	-0,48	-0,18	-0,46	0,13	-	0,18	-0,04	-0,67	-0,55	0,19	0,19	-0,23	X	-0,59	-0,50
magnetfeld_creation.jpg	-0,68	-0,63	-0,33	-0,40	-0,21	0,40	-	X	-0,51	-0,40	0,52	0,52	-0,38	0,16	-0,56	-0,49
magnetfeld_news.jpg	-0,56	-0,53	-0,25	-0,30	-0,14	0,31	X	-	-0,59	-0,44	0,32	0,32	-0,28	0,09	-0,47	-0,40
soldaten_original_creation.jpg	-0,66	-0,20	-0,50	-0,05	-0,52	-0,66	-1,68	-2,25	-	X	-0,22	-0,22	-0,44	-0,22	0,37	0,90
soldaten_original_news.jpg	-0,49	-0,21	-0,68	-0,41	-0,53	-0,30	-0,97	-1,28	0,76	-	-0,40	-0,40	-0,69	0,32	0,31	X
sonnenspiegel_creation.jpg	-0,35	-0,31	-0,20	-0,17	-0,17	0,09	0,16	0,01	-0,13	-0,21	-	X	-0,20	0,10	-0,36	-0,28
sonnenspiegel_news.jpg	-0,35	-0,31	-0,20	-0,17	-0,17	0,09	0,16	0,01	-0,13	-0,21	X	-	-0,20	0,10	-0,36	-0,28
stufenmodell_creation.jpg	-0,14	-0,20	X	0,58	0,67	-0,19	-0,65	-0,64	-0,37	-0,54	-0,41	-0,41	-	0,13	-0,10	-0,19
stufenmodell_news.jpg	-0,53	-0,35	0,11	0,05	0,48	X	-0,63	-0,92	-0,94	-0,48	-0,09	-0,09	0,08	-	-0,58	-0,40
vogelgrippe_creation.jpg	X	0,78	0,07	-0,06	0,07	-0,80	-2,16	-2,21	0,13	0,15	-1,67	-1,67	0,20	0,01	-	0,82
vogelgrippe_news.jpg	-0,26	0,15	-0,29	-0,22	-0,33	-0,54	-1,72	-1,76	0,61	X	-1,10	-1,10	-0,11	0,24	0,72	-

Abbildung 4-24: Ergebnis des kombinierten Vergleichs

Die Trefferquote liegt bei 43,75 %, da 7 von 16 Testbildern dem Vorschaubild aus dem anderen Archiv korrekt zugeordnet werden. Der kombinierte Ansatz bringt folglich keinerlei Besserung mit sich.

4.2.3 Abschließende Bewertung

Die Histogramm-Analyse und die entwickelten Algorithmen zeigen, dass über die Hälfte der Bildpaare identifiziert werden.

Wenn man die Bilder nach dem Distanzwert (Abbildung 4-14) sortiert, erhält man eine Liste, auf der in 62,5 % der Fälle das gesuchte Bild auf dem ersten oder zweiten Platz steht. Berücksichtigt man noch das drittplatzierte Bild beim 2D-Histogramm Vergleich, steigt die Trefferquote auf 87,5 %. Wenn der Rechercheur das im Creationarchiv gefundene Suchergebnis betrachtet, könnte er immer noch relativ

schnell und übersichtlich in einer Vielzahl von Fällen das Gegenstück zum Vorschaubild aus dem Newsarchiv auffinden.

Zur Ergebnisverifikation wurde ein Test durchgeführt, in dem aus einem Stamm von 50 Bildern jedes Mal zufällig 16 Bilder ausgewählt wurden, die mit dem gesuchten Bild verglichen wurden. Dabei stellte sich eine durchschnittliche Trefferquote von 70 % heraus. In diesen Tests wurde die sortierte Liste der Vorschaubilder innerhalb von 3 bis 4 Sekunden ausgegeben; wurde die Datenmenge erhöht, so stieg die Berechnungszeit proportional zur Bildanzahl.

Bei weiteren Tests mit unterschiedlichen Intervallgrenzen (sogenanntes grobes Binning) von Intensitätswerten – Unterteilung in 32 oder 64 Intensitätswerte statt 256 Werte – wurden Trefferquoten unter 50 % erzielt.

Besondere Ausnahmen - wie das folgende Bildpaar - wird man mit dem Vergleich von Farbhistogrammen nicht korrekt zuordnen können:



Abbildung 4-25: Beispiel für nicht auffindbare Bildpaare

Um insgesamt noch bessere Ergebnisse bei den Erstplatzierten zu erzielen, kann man z.B. Gewichtungen für einzelne Farbkanäle (mathematischer Ansatz: multiple lineare Regression) vergeben. So würde bei dem Vergleich von Bildern aus der Nachrichtensendung RTL Aktuell oder Nachtjournal eine stärkere Gewichtung des Blaukanals Sinn ergeben, da die Designvorlage für animierte Grafiken in diesen Sendungen aus Blautönen besteht.

Ein weiterer Verbesserungsvorschlag wäre die Anwendung des Adaptive Binning für Farbhistogramme. Bei dieser Methode werden die Bins nicht festgesetzt. Adaptive

Binning passt sich je nach der aktuellen Distribution der Farben im Bild an und ermöglicht somit unterschiedliche Einteilung der Bins für unterschiedliche Bilder.⁴⁷

Ein darauf basierender Ansatz nennt sich Adaptive Color Histogram (*ACH*) – Verfahren. Diese Methode ermöglicht es Bilder einander zuzuordnen, die sich deutlich sichtbar ähneln, aber unterschiedlich belichtet sind. Grund für die schwerere Zuordnung beim herkömmlichen Vergleich der Histogramme ist das auf der X-Achse nach rechts verschobene Histogramm des heller belichteten Bildes. Der *ACH*-Ansatz löst das Problem in drei Schritten: Berechnen der Histogrammintervalle, Festlegen des Startpunkts eines Histogramms und Messen der Schnittmenge der beiden zu vergleichenden Histogramme.⁴⁸

Ebenfalls basierend auf Adaptive Binning könnte man Kontrastunterschiede in Histogrammen ausgleichen, indem man die betroffenen Histogramme streckt oder staucht und dann miteinander vergleicht.

Zusammenfassend kann man das Ergebnis der Bildvergleiche in der Arbeit positiv und für den Realeinsatz als wirksam einstufen, da es dem Redakteur die Suche nach dem richtigen, grafischen Rohmaterial in vielen Fällen erleichtert.

4.2.4 Integration in die Webanwendung

Aus dem oben genannten Grund wurde das Bildvergleich-Modul auf dem Testserver in die Webanwendung integriert. Ziel dabei war es, die Gebrauchstauglichkeit und die Performance der Komponente zu messen. Zu Ermittlung der Geschwindigkeit wurden Zeitstempel vor wichtigen Programmabschnitten in eine Log-Datei geschrieben.

In der Abbildung 4-26 wird der Ablauf beim Aufruf des Bildvergleichs dargestellt:

⁴⁷ vgl. [LeLi 2004], S.62 f.

⁴⁸ vgl. [YoKiYoKiLe 2006], S.14 f.

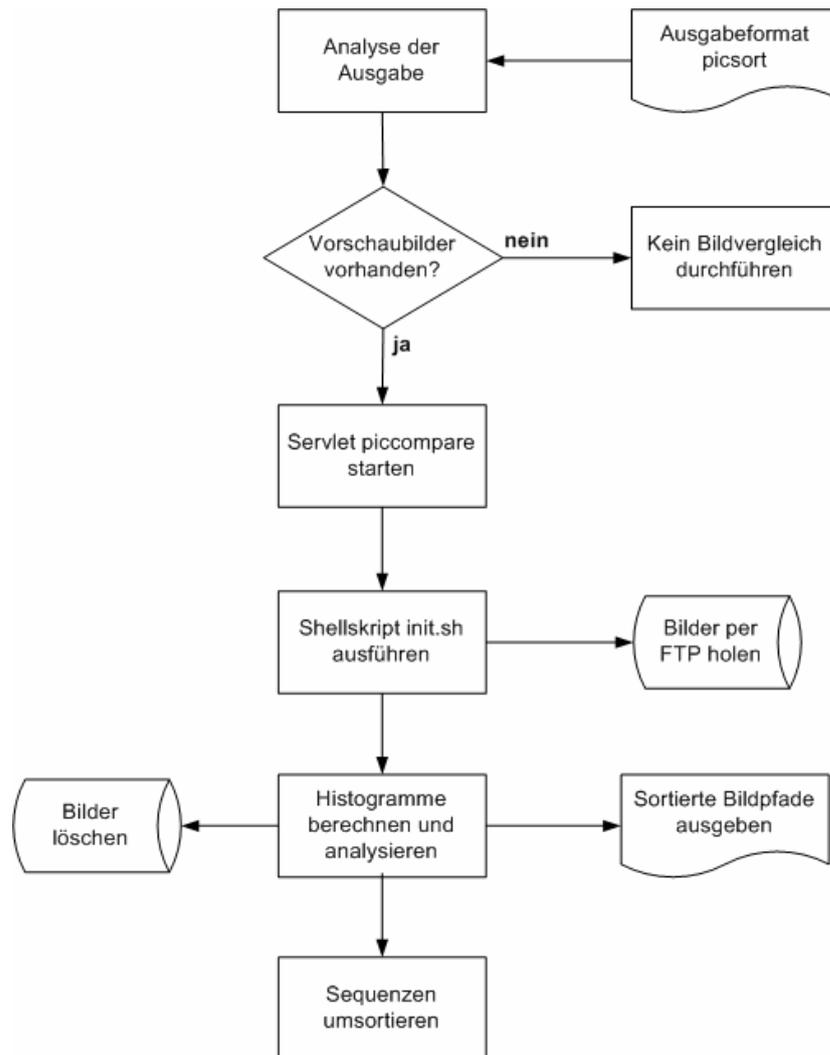


Abbildung 4-26: Datenflußdiagramm Bildvergleich

Der oben dargestellte Datenfluss setzt an der Stelle an, wenn die Suchergebnisse des Creationarchivs im Langformat präsentiert werden.

Zu diesem Zeitpunkt wird im Hintergrund das *TRIP*-Ausgabeformat „picsort“ generiert, welches die benötigten Daten (Dateipfad, Ausstrahlungsdatum, *ID* des Sammelbandes) enthält, um ein Vorschaubild eindeutig zu identifizieren. Wenn das Ausgabeformat keine Informationen enthält wird kein Bildvergleich durchgeführt, ansonsten wird das Servlet „piccompare“ aufgerufen. Dieses Servlet bildet das ursprüngliche Hauptprogramm ab und ruft ein Shellskript namens „init.sh“ auf, welches die Bilder von den verschiedenen Dateiservern in *ein* temporäres Verzeichnis kopiert. Anschließend werden die 2D-Farbhistogramme berechnet und miteinander verglichen. Die Dateipfade der Vorschaubilder werden nach Ähnlichkeit sortiert und in dem versteckten Frame „framehelp“ ausgegeben. Danach wird der temporäre Ord-

ner mit den Kopien der Vorschaubilder gelöscht. Zuletzt werden die Dateipfade aus dem „framehelp“ gelesen und die einzelnen Sequenzen im Langformat zeilenweise nach der sortierten Anordnung vertauscht.

Dieses Verfahren scheint umständlich, ist aber aufgrund der Konfiguration der RTL Server das einzige Vorgehen um Sicherheitsmängel auszuschließen. Der Nachteil hierbei ist der hohe Zeitaufwand; momentan dauert eine Sortierung der Sequenzen je nach Auslastung des Testservers bis zu 7 Sekunden. Dabei benötigt die Initialisierung (Kopieren der Bilder in ein temporäres Verzeichnis) genauso viel Zeit wie der Bildvergleich. Da dieses Ergebnis nicht tragbar ist, wird der Bildvergleich zukünftig nicht automatisch bei der Präsentation der Suchergebnisse durchgeführt, sondern kann auf Wunsch durch Tastendruck gestartet werden. Dieser Kompromiss lässt dem Rechercheur die Entscheidung, ob er die hilfreiche Sortierung benutzen möchte.

4.3 Portierung des Servermoduls *TRIPHighway*

Die Ablösung der Serverkomponente durch eine Eigenentwicklung ist sehr komplex, da sie die zentrale Komponente auf der Serverseite des *TRIPHighway*-Systems darstellt (siehe Abbildung 4-27). Die Serverseite besteht aus einer physikalischen Einheit, auf der sich ein Webserver, ein *TRIP*Server und ein Dateiserver befinden:

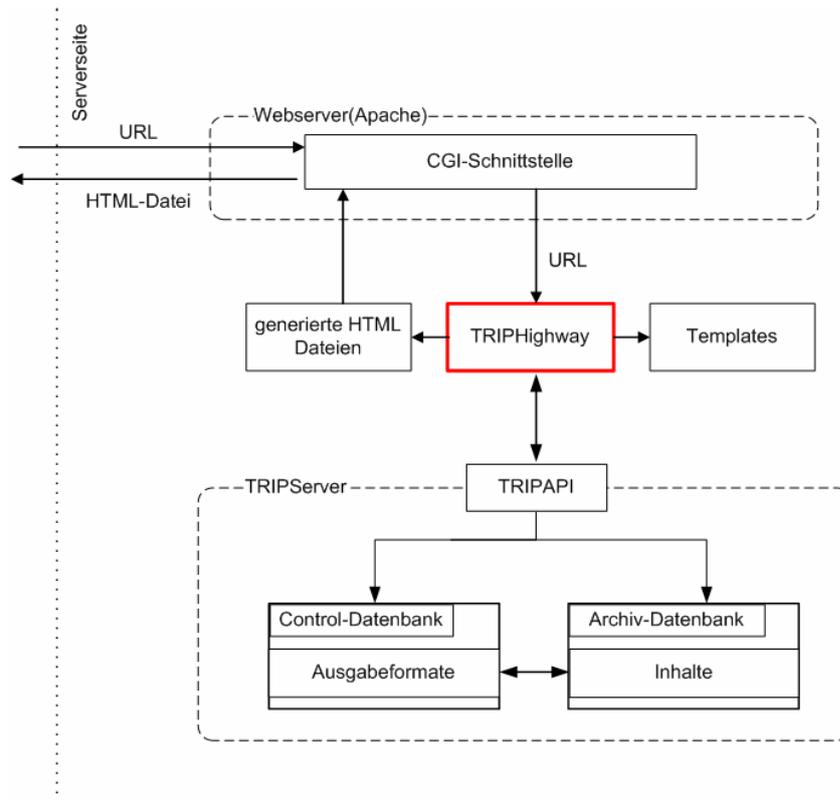


Abbildung 4-27: Portierung TRIPHighway

Um den Aufwand zu bewältigen und alle Probleme zu lösen (siehe Kapitel 3.2.2 und 3.2.3) hat das Projektteam während der Planungsphase die Aufgabenteilung festgelegt. Alle innerhalb der Bachelorarbeit dokumentierten Schritte zur Portierung der Serverkomponente unterstanden meinem Aufgabenbereich und wurden eigenständig implementiert.

Die erstellte Softwarekomponente ermöglicht bereits die archivübergreifende Recherche mit kleinen Einschränkungen.

4.3.1 Funktionen der Eigenentwicklung

Der erste Schritt, den das Modul *TRIPHighway* übernimmt, ist die Analyse der *URL*. Zur Erinnerung: Die *URL* (beinhaltet die zusammengesetzte Suchanfrage aus der Suchmaske) wird von der Webanwendung per Request an den Webserver gesendet (siehe Abbildung 4-27) und die *CGI* nimmt die *URL* entgegen und startet das *TRIPHighway*-Programm. Dieses bekommt die *URL* als Parameter übergeben und kann sie dann analysieren und zerlegen.

Eine Beispiel-URL (der Übersichtlichkeit halber nach jedem ‚&‘-Zeichen durch Zeilenbruch getrennt) sieht folgendermaßen aus:

```

uname=username&
passwd=passwort&
${APPL}=newsarchive8&
${BASE}=NEWS_ARCHIV&
${HILITE}=1&
${HILITEATTR}=<b orangered>&
${CCL}=define THES=PHrase:RTHES.PHrase:TEExt%2BPhrase&
${CCL}=define+view+bilder=topic,sequenz&
${CCL}=define+view+ort=stadt,land&
${CCL}=define+view+titel1=titel,event&
${CCL}=define ,=OR&
${CCL}=define+century+min=1910&
${CCL}=call ase_set_rthw RTLInfo&
${MAXPAGE}=1&
${SNHTML}=hc_err&
${HTML}=hitcount&
${TRIPSHOW}=format=thw_news_titel&
${FREETEXT}=( (schüttel-trauma) AND
(SDATE=(08.2006 TO 09.2006)) AND (VIDEO=$ AND.T VILOC=OL))
AND.T FREI_FUER=(RTLInfo)
NOT.T RECHTE=(Sperre OR Rücksprache OR Kauf)

```

Abbildung 4-28: Beispiel URL TRIPHighway

Die übergebene Zeichenkette (siehe Abbildung 4-28) wird zu Beginn der Verarbeitung durch folgende Schritte analysiert:

1. Die ersten beiden Wertepaare der URL müssen der Username und das Passwort sein, ansonsten wird die Verarbeitung abgebrochen.
2. Alle Pluszeichen ‚+‘ werden durch Leerzeichen ersetzt.
3. Alle dreistelligen Zeichenketten, die mit einem Prozentzeichen beginnen ‚%‘ und auf die ein zweistelliger Hexadezimalwert folgt, werden als Hexadezi-

malwert erkannt und mittels *ASCII*⁴⁹-Tabelle in das reguläre *ASCII*-Zeichen umgesetzt (Bsp.: ‚%2B‘ ersetzt in ein ‚+‘).

In der weiteren Verarbeitung wird die Zeichenkette in die einzelnen Schlüsselwörter (siehe Abbildung 4-29) und die zugehörigen Werte unterteilt. Ein Schlüsselwort ist dabei immer durch ein Dollar-Zeichen ‚\$‘ und geschweifte Klammern ‚{...}‘ umschlossen. Der zugewiesene Wert wird durch ein Gleichheitszeichen ‚=‘ vom Schlüsselwort getrennt. Zusätzlich wird ein neues Syntaxpaar (Schlüsselwort=Wertzuzuweisung) durch das Sonderzeichen ‚&‘ eingeleitet (siehe Abbildung 4-28).

Die Schlüsselwörter werden in der nachfolgenden Tabelle aufgelistet und deren Bedeutung erläutert:

Schlüsselwort	Erläuterung
APPL	gibt den Ordner an, indem die <i>THW</i> Template Dateien auf dem Server gespeichert sind
BASE	Wert beinhaltet die zu verwendende Volltextdatenbank
HILITE	enthält den Wert 0 (keine Hervorhebung) oder 1 (Hervorhebung der Suchbegriffe in der <i>HTML</i> -Ausgabe)
HILITEATTR	enthält die Formatierung für die Hervorhebung in <i>HTML</i> -Syntax
CCL	enthält <i>CCL</i> -Befehle, die an die Datenbank weitergegeben werden
MAXPAGE	enthält die maximale Anzahl der Suchergebnisse, die auf einer Seite ausgegeben werden sollen
SNHTML	enthält den Namen des <i>THW</i> Templates, welches im Fehlerfall zu verwenden ist
HTML	enthält den Namen des <i>THW</i> Templates, welches im Erfolgsfall zu verwenden ist
THWIDS	enthält den Index eines Datensatzes in der aktuellen Suchergebnismenge
TRIPSHOW	enthält das zu benutzende Ausgabeformat

⁴⁹ American Standard Code for Information Interchange

FREETEXT	enthält die eigentliche Suchanfrage an die Volltextdatenbank
----------	--

Abbildung 4-29: Schlüsselwörter in der *TRIPHighway URL*

Bei der Analyse der Syntaxpaare (Schlüsselwort, Wertzuweisung) überprüft die Anwendung, ob der ausgelesene Wert einen *CCL*-Befehl (z.B. find) beinhaltet oder ob der Wert Informationen für die Verarbeitung der Ausgabeformate (z.B. Dateiname des *THW* Templates) enthält.

Ausführen der Suchanfrage. Nach der Analyse der *URL* wird die Suchanfrage an die Datenbank abgesetzt. Um eine Suchanfrage auszuführen, greift die Anwendung auf die *TRIP-API* bzw. auf die in C++ verfassten Wrapper⁵⁰-Klassen der *TRIP-API* zu. Die *API* übergibt die Befehle direkt an den *TRIP*Server, welcher diese dann ausführt und ein Ergebnis in Form eines Resultsets⁵¹ oder im Fehlerfall eine Fehlermeldung zurückgibt.

Im Folgenden werden die vier wichtigsten Methoden der *TRIP-API* vorgestellt, welche von dem Programm aufgerufen werden.

```
bool Login (char *user, char *pwd, char *config);
```

Die Methode registriert einen Anwender für die Verbindung mit der Volltextdatenbank. Neben den Login-Daten wird noch der Pfad einer Konfigurationsdatei übergeben, der die spezifischen Einstellungen des Anwenders lädt.

```
virtual RecordSet *Execute (char *cmd, int maxRecsToRead=-1);
```

Diese Methode führt einen, im ersten Parameter übergebenen, *CCL*-Befehl auf der Datenbank aus. Einige *CCL*-Befehle fragen Daten von der Datenbank ab, andere setzen Werte und geben nichts an die Funktion zurück. *CCL*-Befehle die kein Abfrageergebnis liefern, sollte man als zweiten Parameter eine 0 übergeben, da sonst standardmäßig alle in der Datenbank vorhandenen *IDs* zurückgegeben werden.

```
StrArray *ShowNext (char *outFormName, char *hiliteBegin=NULL,
                    char *hiliteEnd=NULL);
```

⁵⁰ engl. wörtlich: Umschlag. In diesem Fall sind Klassen gemeint, die aus Kompatibilitätsgründen zwischen dem umschlossenen und aufgerufenen Programmcode agieren, um Fehlermeldungen abzufangen.

⁵¹ Klasse, welche die Suchergebnisse der Datenbank beinhaltet und Methoden zur Verfügung stellt, um auf einzelne Ergebnisse zuzugreifen.

Diese Methode gibt jeweils eine Instanz der Klasse StrArray zurück, die jeweils genau einen Datensatz vom gewünschten Suchergebnis enthält. Als erster Parameter wird das gewünschte Ausgabeformat übergeben, indem das Suchergebnis zurückgegeben werden soll. Im zweiten und dritten Parameter befinden sich *HTML*-Elemente in Form von Zeichenketten (wie z.B. „“ und „“), welche eingegebene Suchbegriffe bei der Ausgabe zusätzlich formatieren.

```
long GetLastErrorCode ()
```

Diese Methode verwendet man, um mögliche Fehlercodes abzufangen und dem Benutzer sprechende Fehlermeldungen auszugeben.

Volltextdatenbanksprache CCL. Zum weiteren Programmverständnis werden auch die grundlegenden *CCL*-Befehle erklärt, die in einer übergebenen *URL* enthalten sind und ohne große Änderungen an die *TRIP-API* übermittelt werden, um eine Suchanfrage auf der Volltextdatenbank auszuführen.

```
base [database]
Bsp.: base NEWS_ARCHIV
```

Der base-Befehl öffnet eine oder mehrere Datenbanken über die gesucht werden soll. In diesem Beispiel öffnet er die Datenbank des Newsarchivs.

```
define view [view name]=[field,field,...]
Bsp.: define view ort=stadt,land
```

Der define-Befehl setzt Standard (Default)-Einstellungen für die momentane *TRIP* Session (engl. Sitzung). Hier wird der „define view“-Befehl erläutert, der es ermöglicht verschiedene Datenbankfelder zu kombinieren und die Suche über eine View, statt über die einzelnen Felder durchzuführen. In diesem Beispiel definiert er die View Ort, als eine Zusammensetzung der Felder Stadt und Land.

```
find [search condition]
Bsp.: find (schüttel-trauma) AND (SDATE=08.2006)
```

Dieser Befehl enthält die eigentliche Suchanfrage. Alle einzelnen Suchbegriffe werden durch die boolesche Algebra (AND, OR, XOR, NOT) verknüpft. Soll nicht über den Volltext gesucht werden, sondern über ein spezielles Feld, so muss dies explizit angegeben werden (z.B.: FREI_FUER=(RTLInfo)). Im obigen Beispiel werden alle

Datensätze gesucht, die den Volltext „schüttel-trauma“ und das Sendedatum „08.2006“ enthalten.

Generieren der *HTML*-Dateien. Nachdem eine Suchanfrage an die Datenbank gestellt wurde, erhält das Programm von der jeweiligen Volltextdatenbank das Suchergebnis in Form einer Instanz der Klasse TripRecordSet zurück. Die darin enthaltenen Metadaten werden in dem nächsten Verarbeitungsschritt dazu verwendet, um vorliegende *THW* Templates zu füllen. Den Namen des zu füllenden *THW* Templates erhält die Anwendung aus dem Schlüsselwort *HTML*, welches vorher bereits in der Syntaxanalyse der *URL* ausgelesen wurde (siehe Abbildung 4-29).

Anschließend wird die Template-Datei eingelesen und es wird - wie im ersten Schritt der Verarbeitung - nach Schlüsselwörtern gesucht, die analog zu den Schlüsselwörtern der *URL* aufgebaut sind (Bsp.: $\${WORT}$), in diesem Fall allerdings als Platzhalter für Metadaten stehen:

Schlüsselwort	Erläuterung
CURRBASE	Platzhalter für die momentan verwendete Volltextdatenbank
CURRID	Platzhalter für die <i>ID</i> des aktuell gesuchten Datensatzes
CURRIS	Platzhalter für die Nummer des aktuellen Datensatzes im Suchergebnis
RECORDS	Platzhalter für die Trefferanzahl
TRIPFORM	Platzhalter für die Metadatenausgabe im formatierten Zustand
TRIPSLI	Platzhalter für die <i>URL</i> , welche die Suchanfrage für die momentan angezeigten Datensätze enthält
FINDSTR	Platzhalter für die Zeichenkette, welche die Suchbegriffe enthält
DPYTERM	Platzhalter für die Anzahl der im Suchfeld Form oder Redakteur eingegebenen Inhalte
MATCHES	Platzhalter für die Anzahl angezeigter Suchbegriffe
DBERROR	Platzhalter für die Datenbank-Fehlermeldung
PREVREC	Platzhalter für die <i>URL</i> , welche die Suchanfrage für den nächsten anzuzeigenden Datensatz enthält

NEXTREC	Platzhalter für die <i>URL</i> , welche die Suchanfrage für den vorherigen anzuzeigenden Datensatz enthält
FORWARD	Platzhalter für die <i>URL</i> , welche die Suchanfrage für die nächsten anzuzeigenden Datensätze enthält (Die Menge wird durch MAXPAGE bestimmt)
BACKWARD	Platzhalter für die <i>URL</i> , welche die Suchanfrage für die vorherigen anzuzeigenden Datensätze enthält (Die Menge wird durch MAXPAGE bestimmt)
Bedingung/Schleife	Erläuterung
TRIPIF PREVRIS	Beginn der IF-Bedingung (wird als wahr erkannt, wenn es vor diesem Datensatz einen weiteren anzuzeigenden Datensatz gibt)
TRIPIF MOREIS	Beginn der IF-Bedingung (wird als wahr erkannt, wenn es nach diesem Datensatz einen weiteren anzuzeigenden Datensatz gibt)
TRIPIF END	Ende der IF-Bedingung
TRIPLOOP LOOPTYPE="ris"	Beginn der Schleife (wird so oft durchlaufen, wie Datensätze vorhanden sind)
TRIPLOOP END	Ende der Schleife

Abbildung 4-30: Schlüsselwörter in den THW Templates

Im unteren Teil der Abbildung 4-30 sind die Schleife und die IF-Bedingung aufgeführt, die ebenfalls in einer Template-Datei enthalten sein können und einen beliebigen und formatierten *HTML*-Code umschließen dürfen. Die Syntax dieser speziellen Schlüsselwörter unterscheidet sich gegenüber den Anderen, in dem diese nicht durch ein Dollarzeichen ‚\$‘ und geschweifte Klammern ‚{...}‘ umschlossen sind, sondern durch spitze Klammern ‚<...>‘ eingegrenzt werden.

Der relevanteste Teil der Template-Datei „thw8_title.thw“, der für die Metadaten-Ausgabe im Titelformat zuständig ist, sieht folgendermaßen aus:

```
<form name="formText">
<table id=tblRecords valign=top border=0>
  <th colspan=8 align=left">
    <font color=#000000>Anzahl der Datensätze: ${RECORDS}</font>
```

```

</th>
<tr>
  <td></td>
  <td><font color=#000000>Nr.</font></td>
  <td><font color=#000000>A</font></td>
  <td><font color=#000000>DB</font></td>
  <td><font color=#000000>ID</font></td>
  <td><font color=#000000>TC-Anfang</font></td>
  <td><font color=#000000>Titel</font></td>
</tr>
#{TRIPFORM}
</table>
</form>
<form name="formData">
  <input type=hidden name="docID" value="OutputFormTitle">
  <input type=hidden name="prevURL"
    value="<TRIPF PREVRIS>#{BACKWARD}<TRIPF END>">
  <input type=hidden name="nextURL"
    value="<TRIPF MORERIS>#{FORWARD}<TRIPF END>">
  <input type=hidden name="varNRecords" value="#{RECORDS}">
  <input type=hidden name="tripsli" value="#{TRIPSLI}">
  <input type=hidden name="currRIS" value="#{CURRRIS}">
</form>

```

Abbildung 4-31: Beispiel Template-Datei TRIPHighway

Nachdem die Platzhalter durch alle Datenbankinhalte des Suchergebnisses ersetzt wurden, wird die entstandene *HTML*-Datei als Serverantwort an den Browser gesendet. Tritt bei der Suchanfrage oder der Template-Verarbeitung ein Fehler auf, so wird eine Error-Template-Datei für die Fehlerausgabe aufbereitet und per *HTML* Response zum Browser gesandt. Der Name der Error-Template-Datei wird auch mit in der *URL* - im Schlüsselwortpaar SNHTML - übergeben und bei der Syntexanalyse extrahiert. Sollte kein Name angegeben sein, wird die Standard Error-Datei „trip_err.thw“ für die Fehlerausgabe verwendet.

4.3.2 Aufgetretene Probleme und Abgrenzung

Das Modul wurde mit der IDE Microsoft Visual Studio 2005⁵² entwickelt und getestet. Zusätzlich erfolgten regelmäßig Tests auf dem Testserver, welcher mit dem Betriebssystem HP-UX arbeitet.

Größere Schwierigkeiten, welche die Entwicklung des Servermoduls stark beeinträchtigten, traten in der Entwicklungsphase nicht auf. Trotzdem gab es ein paar zeitverzögernde Probleme bei der Implementierung:

- Zwischen der Windows und der UNIX *TRIP-API* gibt es gravierende Unterschiede in Bezug auf die einzelnen Rückgabewerte im Erfolgs- und Fehlerfall. Die unterschiedlichen Returncodes werden durch Wrapper-Klassen abgefangen.
- Sobald die Serverkomponente über die *CGI* des Apache Webservers gestartet wird, sind die UNIX-Umgebungsvariablen nicht mehr gesetzt. Diese müssen somit eigenständig durch das Modul *TRIPHighway* festgelegt werden, damit z.B. wichtige Pfadangaben für die weitere Programmverarbeitung bekannt sind.
- Das von TietoEnator stammende *TRIPHighway*-Modul speichert die Suchergebnisse zusätzlich zur jeweiligen Ausgabe in temporären Dateien. Um diesen Zeitaufwand zu verringern gibt die Eigenentwicklung immer nur die Trefferanzahl aus, die durch den Wert hinter dem Schlüsselwort *MAXPAGE* definiert wird und somit im jeweiligen Ausgabeformat dargestellt werden soll. Die anderen Suchergebnisse werden dynamisch nachgeladen, sobald der Benutzer in der Navigationsleiste vor- oder rückwärts blättert.

Die Portierung durch eine Eigenentwicklung ist abgeschlossen. Die archivübergreifende Recherche kann mit Hilfe des neuen Moduls durchgeführt werden. Nun kann das neue Modul auf der Spiegelung des Produktivservers installiert und einem Langzeittest unterzogen werden.

Die Entwicklung einer Datenbank-Schnittstelle, um beispielsweise über ein *ODBC*⁵³-System auf eine Sybase oder Oracle-Datenbank zuzugreifen, wurde nicht implemen-

⁵² weitere Informationen im Kapitel 6.2.1 „Microsoft Visual Studio 2005“

⁵³ Open Database Connectivity

tiert, da es die derzeitige Konfiguration der RTL *HP*⁵⁴-Server nicht ermöglicht, ein *ODBC*-System zu installieren. Ein theoretisches Konzept hierfür wurde aber entworfen.

4.3.3 Konzept für ein erweitertes DB-System

Falls die Server von RTL zukünftig ausgetauscht werden (etwa beim Umzug nach Köln-Deutz 2008), kann ein *ODBC*-System, beispielsweise mit Linux-Servern, erfolgreich installiert und konfiguriert werden. Das TRIPHighway Modul wäre dann durch eine Erweiterung in der Lage, die Kommunikation mit mehreren Datenbanken zu steuern; aus diesem Grund erfolgt neben der Funktionserweiterung auch eine Namensänderung (siehe Abbildung 4-32):

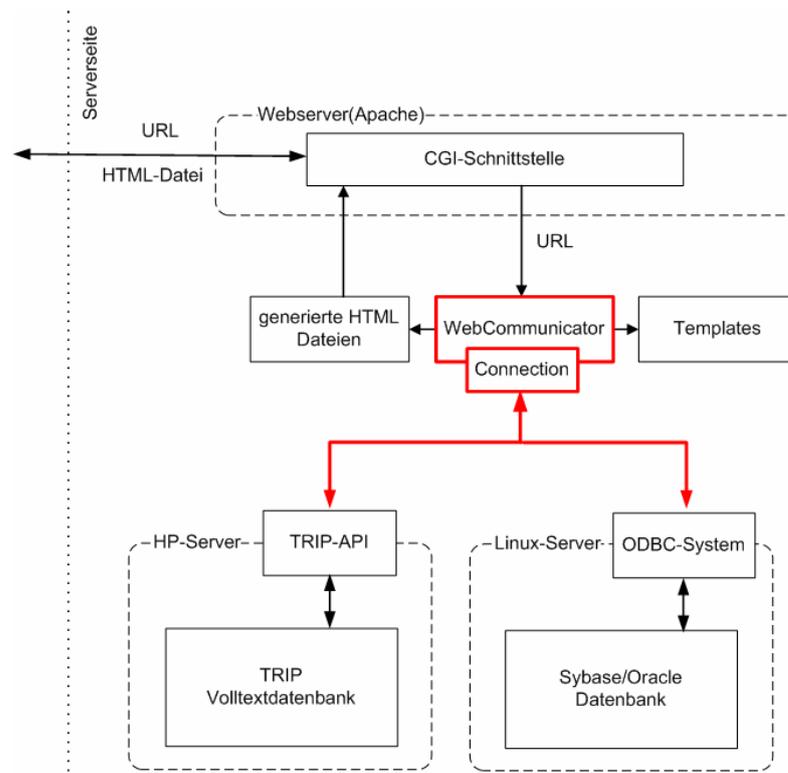


Abbildung 4-32: Beispielkonzept für Datenbankanbindung

In dieser Systemarchitektur wird das eigenentwickelte Modul (WebCommunicator) weiterhin die übergebene *URL* analysieren.

Je nachdem welcher Wert hinter dem Schlüsselwort *BASE* steht, wird eine entsprechende Instanz der Klasse *Connection* (siehe Abbildung 4-33) erzeugt, die entweder

⁵⁴ Hewlett-Packard

eine Verbindung zur *TRIP*-Volltextdatenbank oder zu einer über *ODBC* angesteuerten Datenbank aufbaut.

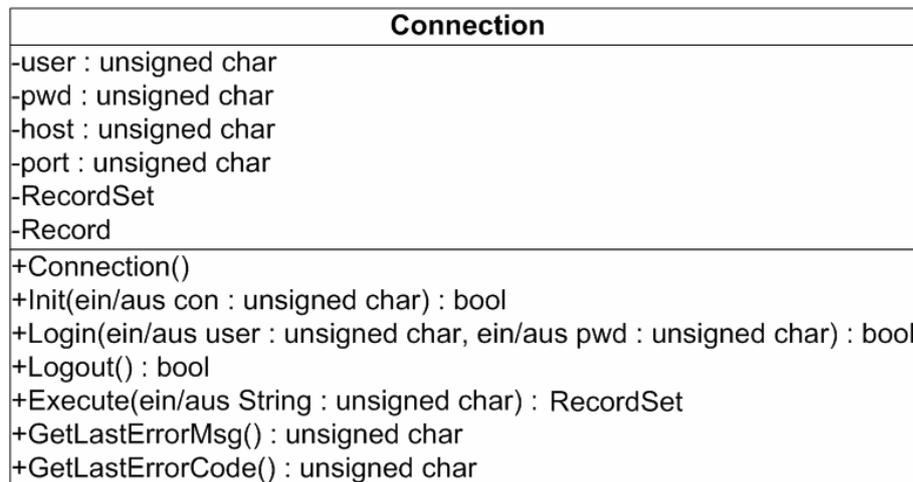


Abbildung 4-33: Klasse Connection

Die Connection-Klasse beinhaltet Methoden zum Verbindungsauf- und Verbindungsabbau (Login, Logout), die Kernmethode zum Ausführen einer Suchanfrage (Execute), welche die Suchergebnisse in Form eines RecordSets zurückgibt und die Methoden zur Fehleranalyse (GetLastErrorMsg, GetLastErrorCode). Der Methode Init wird nur ein Connection-String übergeben, so dass die Flexibilität beim Zugriff auf die verschiedenen Datenbanken gewährleistet ist.

Der große Vorteil bei diesem Konzept ist, dass Fremdsysteme, wie z.B. Archive von anderen, zur RTL Gruppe hinzugehörenden Unternehmen, in den Programmablauf mit eingebunden werden können, ohne die Struktur der fremden Datenbanken anzupassen oder die Inhalte womöglich zu exportieren, um sie danach in einer *TRIP* konformen Datenbankstruktur in die Volltextdatenbank zu integrieren.

Heterogene Datenbanksysteme können mit dem neuen WebCommunicator-Modul kombiniert, verarbeitet und deren Datenbestände in einer Weboberfläche aufbereitet und visualisiert werden ohne Datenbestände grundlegend zu verändern.

Eine weitere Vision ist das Verarbeiten der Suchanfrage und die gleichzeitige Aufbereitung der Metadaten in einer *XML*-Datei, welche beispielsweise nach den Vorgaben

des Protokolls *MOS*⁵⁵ (Media Object Server) strukturiert ist. In dieser Form stehen die Informationen in einem unabhängigen Format zur Verfügung und können direkt in die nächste Softwarekomponente des Arbeitsablauf (z. B. Videoserver) integriert werden.

⁵⁵ *MOS* ist ein sieben Jahre altes Protokoll, welches die Kommunikation zwischen Newsroom Computersystemen und Media Object Servern ermöglicht. Es wird aus einem Zusammenschluss von Herstellern, Softwareentwicklern und Endbenutzern entwickelt und gewartet. (www.mosprotocol.com)

5 Zusammenfassung und Ausblick

Die Optimierung des Metadatenmanagements zur verbesserten Archiv-Recherche ist erfolgreich abgeschlossen.

Nach einer aufwendigen Problemanalyse durch mehrere involvierte Auftraggeber und das vielseitige Aufgabenfeld wurden viele heterogene Lösungsansätze im Soll-Konzept, sowohl für die schnittstellenübergreifende Suche als auch für die Ablösung des kostenpflichtigen Servermoduls *TRIPHighway*, entwickelt.

Durch die Realisierung der Anforderungen ist ebenfalls eine Suche nach produziertem Material aus dem Bereich Information von RTL Creation möglich. Dieser Schritt stellt eine erhebliche Arbeitserleichterung für den Redakteur und eine Aufwandsentlastung für die Archivare dar. Der Rechercheur hat in der Webanwendung die Wahl zwischen automatischer und selbständiger Archivsuche.

Der Bildvergleich auf Basis von Farbhistogrammen wurde in Form von vier mathematisch verschiedenen Ansätzen implementiert und getestet. Dabei hat der Ansatz „Vergleich von 2D-Farbhistogrammen“ die beste Trefferquote (87,5 % - gesuchtes Bild auf einem der ersten drei Plätze) erzielt. Der Algorithmus kann somit erfolgreich zur Hervorhebung der gesuchten Sequenz und damit zur verbesserten Archivrecherche beitragen.

Positiv zu bewerten ist ebenso die Entwicklung und Integration der neuentwickelten Serverkomponente *TRIPHighway*. Alle Verarbeitungsstufen (Syntaxanalyse der *URL*, Zusammensetzen der Suchabfrage, Generierung der *HTML*-Dateien) sind erfolgreich umgesetzt worden. Die Performance beim Zugriff auf die Volltextdatenbank wurde durch das dynamische Nachladen der Suchergebnisse verbessert. Aufgrund der Konfiguration der RTL-Server konnte allerdings keine erweiterte Datenbank-Schnittstelle entworfen werden; dafür wurde aber ein Konzept für ein solches Vorhaben erstellt.

Bis Ende August 2007 werden die drei einzelnen Komponenten zusammengeführt und anschließend einem Langzeittest unterzogen. Damit ist die Basis für ein Metadaten-Netz entstanden. Die Flexibilität der Eigenentwicklung und die firmen- und archivübergreifende Suche ermöglichen ein noch schnelleres und qualitativeres Produzieren von Nachrichten- und Informationsbeiträgen, als es heute schon Realität ist.

6 Anhang

6.1 Begriffsbestimmungen und Definitionen

In diesem Kapitel werden die in der Arbeit verwendeten Fachbegriffe erklärt, näher bestimmt und von verwandten Begriffen abgegrenzt.

6.1.1 Ampelsystem

Das Ampelsystem ist im Zuge der Rechteverwaltung von *TRIP* im November 2006 entwickelt worden. Die Aufgabe des Systems ist es, die einzelnen Sequenzen eines Beitrags farblich mit Hilfe der Ampelfarben (rot, orange, grün) zu kennzeichnen, um dem Rechercher die Rechtlage für den ausgewählten Bereich anzuzeigen (siehe Abbildung 6-1).

12:00:44:15 1 sec. ATVP Kauf		Österreich / *	Archiv	Ludwig Koch, Vater von Natascha Kampusch , weint
12:00:46:01 14 sec. RTL MÜ frei		Österreich / *	PK	OT Psychiater Max Friedrich . O-Ton : gegenwärtig fühle ich mich an meinem Aufenthaltsort wohl, vielleicht ein bisschen bevormundet, beschlossen, nur telefonische Kontakte zu Familie aufnehmen zu wollen
12:01:00:15 3 sec. ORF Sperr		Österreich / *	Archiv, Foto	Natascha Kampusch , als Kind/Mädchen

Abbildung 6-1: Ampelsystem am Beispiel eines Beitrags über "Natascha Kampusch"

Dieses Beispiel ist aus Sicht des Bereiches RTL Information gewählt; die Rechtlage für die gleichen Sequenzen kann für einen anderen Bereich (z.B. RTL II, VOX, n-tv) ganz anders aussehen. Deshalb muss der Suchende zu Beginn der Recherche den Bereich auswählen, für den er Sequenzen sucht.

Bei der Präsentation der Suchergebnisse gibt es diese drei Kennzeichnungen:

- Kaufmaterial (orange Farbe) wurde einmalig z.B. von einer Agentur für einen bestimmten Preis eingekauft und darf einmalig ausgestrahlt werden. Wenn eine Redaktion diese Sequenz in einem Beitrag wiederverwenden möchte, muss sie erneut den Kaufpreis zahlen.
- Freies Material (grüne Farbe) darf immer wieder verwendet werden.

- Gesperrtes Material (rote Farbe) darf in keinem Fall wiederverwendet werden. Wird es dennoch in einem Beitrag wiederverwendet, führt dies zu hohen Geldbußen.

6.1.2 Broadcast

Im Zusammenhang mit *TRIP* beschreibt Broadcast (engl. Ausstrahlung) die Qualität einer Sequenz. Liegt diese in Broadcast-Qualität vor, so liegt sie in einer hohen Auflösung (HiRes) auf dem Archiv-Server und kann in dieser Qualität direkt in den neuen Beitrag hineingeschnitten werden.

Das Derivat zur Broadcast-Qualität ist die Preview-Qualität. Hier liegt das Material in einer niedrigen Auflösung (LoRes) vor und kann nur zur Sichtung des Archivmaterials verwendet werden. Soll die ausgewählte Sequenz in einen neuen Beitrag integriert werden, muss sie zuerst von einem Archiv-Band eingelesen werden.

6.1.3 VPMS PreviewClient

Dieser von *S4M* entwickelte Videoplayer (Preview Client) ist eine zentrale Komponente des Video Management Production Studios (*VPMS*) und dient bei RTL zur Anschauung, Vorerfassung und zum Schnitt von Videomaterial.

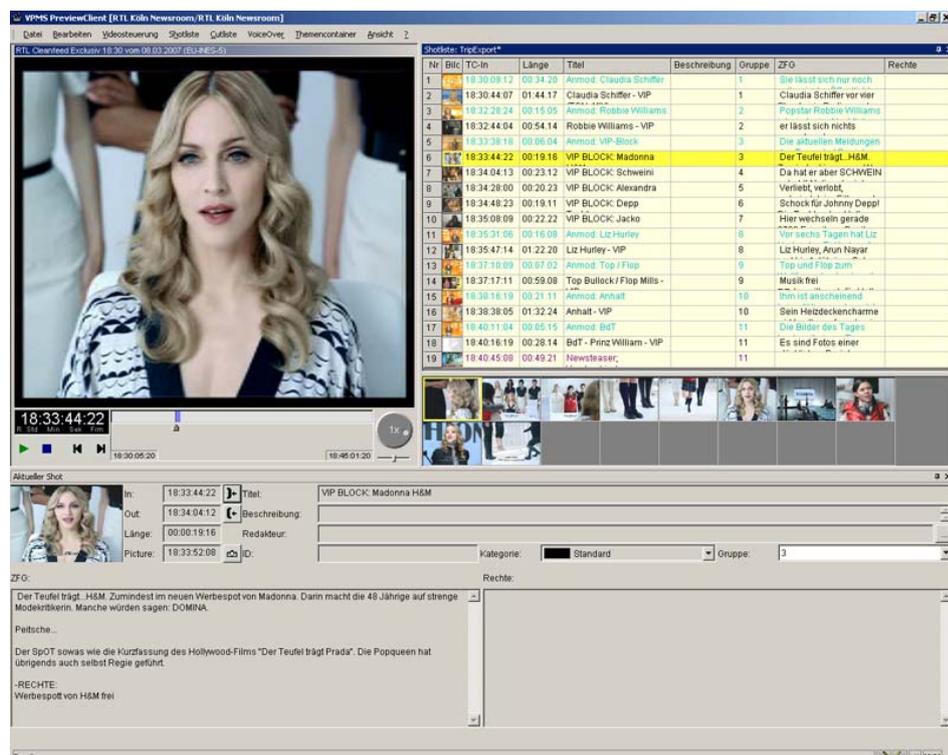


Abbildung 6-2: Vorerfassung eines Beitrags durch VPMS PreviewClient

6.1.4 Spitzen und Löcher

In der digitalen Bildverarbeitung trifft man bei Histogrammen auf isolierte Spitzen und Löcher.

Spitzen beschreiben eine hohe, lokale Häufigkeit von Intensitätswerten gegenüber den benachbarten Werten; Löcher sind genau das Gegenteil. Zum besseren Verständnis ist hier ein Histogramm abgebildet, welches Spitzen (roter Pfeil) und Löcher (blauer Pfeil) beinhaltet:

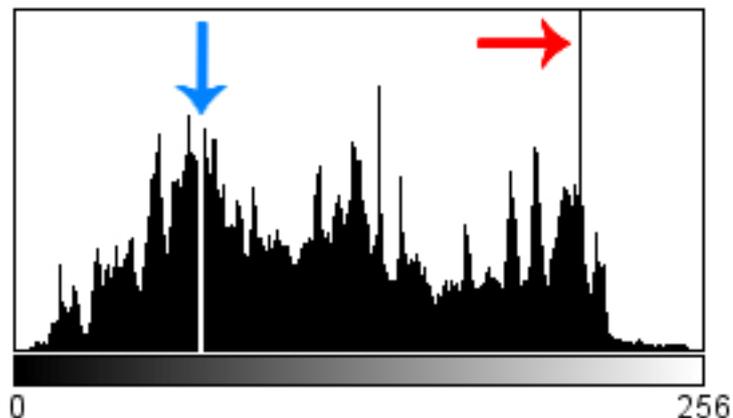


Abbildung 6-3: Spitzen und Löcher im Histogramm

6.2 Entwicklungswerkzeuge

6.2.1 Microsoft Visual Studio 2005

Microsoft Visual Studio 2005 ist die momentan aktuelle, integrierte Entwicklungsumgebung der Firma Microsoft.

Die Software bietet umfassende Möglichkeiten zur Erstellung von klassischen Windows-Clientanwendungen, interaktiven Webanwendungen und datengesteuerten Anwendungen mit den Programmiersprachen Visual C++, Visual C#, Visual Basic und Visual J#.

Zur Sicherstellung der Programmlauffähigkeit wurde die Plattform .NET Framework für klassische Windowsanwendungen und die reduzierte Plattform .NET Compact Framework für mobile Endgeräte entwickelt.⁵⁶

⁵⁶ vgl. [MicroVisual 2007]

Die Software wird in diesem Projekt verwendet, da sie das Standard – Entwicklungswerkzeug bei der *CBC* ist. Viele C++ Projekte wurden bereits mit Hilfe der Entwicklungsumgebung realisiert. Zusätzlich kooperiert sie mit dem Verwaltungssystem Microsoft Visual Source Safe, welches die Standardsoftware für teambasierte Entwicklung bei der *CBC* darstellt.

6.2.2 Microsoft Visual Source Safe

Das von Microsoft entwickelte Versionskontrollsystem kann sämtliche Dateitypen (Quellcode, Dokumentationen, Binärdateien) eines Projektes in einer Datenbank verwalten. Die Hauptaufgabe des Systems ist es, nur einem Entwickler auf einer Datei den Schreibzugriff zu erlauben, so dass der Quellcode nur von einer Person geändert werden darf, bis die Datei wieder „eingescheckt“ (neue Version in die Datenbank geladen) wird.

Durch den Eincheckvorgang werden eine neue Datei und eine neue Dateiversion erstellt, so dass der alte Stand ebenfalls in der Datenbank erhalten bleibt. Jeder Entwickler aus dem Team kann die letzte oder auch eine vorherige Version der Datei abrufen und wieder editieren.

Ein besonderer Vorteil bietet die Integration in das Visual Studio 2005, da dem Programmierer ein Großteil der Arbeit mit der Versionierung abgenommen wird.⁵⁷

6.2.3 Primalscript

Der von der Firma Sapiens entwickelte Scripteditor kann aufgrund des großen Funktionsumfangs schon als Entwicklungsumgebung bezeichnet werden. Er unterstützt bekannte Scriptsprachen wie VBScript, KiXtart, ActionScript und Javascript. Zusätzlich wird die Arbeit mit *HTML*- und *XML*-Dateien durch die farbliche Syntaxhervorhebung und durch viele eingebaute Werkzeuge vereinfacht.⁵⁸

Der Editor wird in diesem Projekt für das Erstellen der *HTML*-Seiten und der Javascript-Funktionalitäten verwendet, da er eine sehr übersichtliche Code-Darstellung bietet.

⁵⁷ vgl. [MicroSource 2007]

⁵⁸ vgl. [Sapiens 2007]

6.2.4 ImageJ

Diese lizenzfreie Bildverarbeitungssoftware hat Wayne Rasband mit der Programmiersprache Java entwickelt. Die plattformunabhängige Software kann sowohl als Java Applet im Internet oder auch als herunterladbare Anwendung auf jedem PC, der eine Virtuelle Maschine mit der Java Version 1.4. oder höher installiert hat, eingesetzt werden.⁵⁹

ImageJ bietet einerseits bereits fertige Werkzeuge zur Darstellung und interaktiven Manipulation von Bildern, andererseits lässt es sich durch eigene Softwarekomponenten erweitern. Diese Erweiterungen werden in Form von Java-Modulen (sogenannte Plugins) entwickelt und können direkt im Programm übersetzt und ausgeführt werden.⁶⁰

In diesem Projekt wird ImageJ zur Bildanalyse und zum Bildvergleich der Grafiken aus dem Creation- und Newsarchiv eingesetzt.

6.3 Eingesetzte formale Sprachen

6.3.1 C++

Diese hybride Programmiersprache unterstützt sowohl die prozedurale, als auch die objektorientierte Programmierung. Sie wurde ursprünglich 1979 von Dr. Bjarne Stroustrup entwickelt, um Simulationsprojekte mit geringem Speicher- und Zeitbedarf zu programmieren.

Die Basis der Entwicklung war die Programmiersprache C, so dass es aufgrund der Abwärtskompatibilität möglich ist, alle C-Funktionen in C++ Programmen zu verwenden. Der momentane *ISO/IEC* Standard der Sprache lautet 14882: 2003.⁶¹

C++ wird schon seit Beginn des Projektes *TRIP* bei RTL für die Programmierung eines Clients eingesetzt, welcher mit der Volltextdatenbank über eine in C verfasste Application Programming Interface (*API*) kommuniziert und *CCL*-Befehle an die Volltextdatenbank absetzt und die Ergebnisse verarbeitet. Die lange Erfahrung des

⁵⁹ vgl. [ImageJ 2007]

⁶⁰ vgl. [BurBur 2006], S. 28 - 29

⁶¹ vgl. [Wolf 2006], S. 25-27

Projektteams und die Geschwindigkeit bei der Verarbeitung von Befehlen führen dazu, dass sie auch in diesem Projekt eingesetzt wird.

6.3.2 CCL

Die Common Command Language (*CCL*) ist eine Kommandosprache für Informationsbeschaffung (Information-Retrieval)-Systeme. Sie wurde 1993 durch den *ISO*-Standard 8777 festgelegt und versucht Suchkommandos in Datenbanken zu vereinheitlichen, so dass der Benutzer nicht für jede Datenbank eine neue Sprache erlernen muss.⁶²

Die *CCL* kommt ohne grafische Benutzeroberflächen aus, fordert somit aber gleichzeitig genaue Kenntnisse über deren Syntax, um eine qualitative Trefferliste zu erhalten.⁶³

Bei *RTL* wird die Kommandosprache verwendet, da nur mit dieser eine Informationsbeschaffung aus der Volltextdatenbank *TRIP* möglich ist.

6.3.3 HTML

Die vom Web-Gründer Tim Berners-Lee entwickelte Hyper Text Markup Language (*HTML*), dient dazu Texte zu strukturieren. Egal ob man Überschriften, Textabsätze, Listen oder Tabellen erzeugen möchte, es ist vieles möglich, um die Informationen im Internet zu präsentieren.

Nicht-textuelle Inhalte, wie z.B. Grafiken oder Dateiquellen kann man durch anklickbare Verweise (sogenannte Links) in die Webseiten einbinden. Ebenso bietet die Auszeichnungssprache noch Schnittstellen zu Erweiterungssprachen wie Cascading Style Sheets (*CSS*) oder Javascript an, mit denen man die Interaktion mit dem Anwender steuern kann.⁶⁴

Ein weiteres Angebot von *HTML* sind Formen, die zwischen dem Text einer Seite integriert werden und dem Benutzer zur Verfügung stehen um Daten einzugeben oder wie in der *TRIP*Highway-Anwendung Suchabfragen zu formulieren.

⁶² vgl. [Garman 1995]

⁶³ vgl. [Saremba 1999]

⁶⁴ vgl. [Selfhtml 2007]

6.3.4 JavaScript

Die von Netscape entwickelte und lizenzierte Programmiersprache Javascript dient dazu die Maus- und Tastatureingaben des Anwenders einer Webseite entgegen zu nehmen, zu verarbeiten und darauf mit dynamischen Änderungen innerhalb der angezeigten Webseite zu reagieren. Beispiele dafür sind die Überprüfung der Eingaben auf Vollständigkeit und Plausibilität in einem *HTML*-Formular oder die Überprüfung, welcher Browser vom Benutzer gerade verwendet wird.

Sobald eine *HTML*-Seite einmal in den Hauptspeicher des PCs geladen wurde, kann der Javascript-Code ohne Webserver ausgeführt werden.

Grundlage beim Zugriff auf die *HTML*-Elemente liefert dabei die Programmierschnittstelle „Document Object Model“ (*DOM*)-Schema.⁶⁵

6.3.5 Java

Die objektorientierte Programmiersprache wurde von Sun Microsystems im Rahmen des „The Green Project“ entwickelt.

Die größten Vorteile der Sprache und der daraus entstandenen Technologie sind die Plattform- und damit Geräteunabhängigkeit, Vielseitigkeit, Effizienz und Sicherheit. Mittlerweile wird sie auf mehr als 2,5 Milliarden Geräten unterstützt.⁶⁶

Die Sprache wird in diesem Projekt im Zusammenhang mit der Software ImageJ verwendet. Es sollen Java-Module implementiert werden, welche die Bilder des Creation- und Newsarchivs analysieren und vergleichen.

⁶⁵ vgl. [SelfhtmlJ 2007]

⁶⁶ vgl. [Java 2007]

Literaturverzeichnis

Bücher

- [BurBur 2006] *Burger, W.; Burge, M.:* Digitale Bildverarbeitung – Eine Einführung mit Java und ImageJ, 2. Aufl, Berlin: Springer Verlag, 2006. – ISBN 3-540-30940-3
- [LuNiKa 2004] *Lugmayr, A; Niiranen, S.; Kalli, S.:* Digital Interactive TV and Metadata. Future Broadcast Multimedia (Signals and Communication Technology), 1. Aufl, Berlin: Springer Verlag, 2004. – ISBN 0-38720-843-5
- [Steffens 2004] *Steffens, U.:* Metadaten-Management für kooperative Anwendungen, 1.Aufl. Berlin: dissertation.de – Verlag im Internet GmbH, 2004. – ISBN 3-86624-012-0
- [Wolf 2006] *Wolf, Jürgen.:* C++ von A bis Z, 1. Aufl, Bonn: Galileo Press Verlag, 2006. – ISBN 3-898-42816-8

Artikel aus Fachzeitschriften / internen Dokumentationen

- [Ebner 2005] *Ebner, A.:* Austausch von Metadaten – Broadcast Metadata exchange Format, BMF In: Aufbau des Datenmodells. Institut für Rundfunktechnik (2005), Nr. B 193 /2005, S. 7
- [Garman 1995] *Garman, N.:* What ever happend to common command language?, In: Online (November 1995), S. 7-8
- [HeKuSaSi 2006] *Heidekum, A.; Kupke, R.; Saremba, D.; Simons, C.:* Zielarchitektur Metadatenmanagement. In: TRIP (Köln 2006), Interne RTL-Dokumentation, v1.0
- [ISO 2003] *International Organization for Standardization (ISO).:* Information technology –Metadata registries (MDR) – Part 3. In: Registry metamodel and basic attributes. In-

ternational Standard ISO/IEC 11179-3 (Genf 2003),
Second edition, S. 7

- [LeLi 2004] Leow, W.; Li, R.: The analysis and applications of adaptive-binning color histograms In: Computer Vision and Image Understanding, Special Issue: Colour for Image Indexing and Retrieval, Vol. 94, Issues 1-3. National University of Singapore. (Singapore 2004).
- [Saremba 1999] Saremba, D.: Einleitung, In: Dokumentationssystem TRIP (Köln 1999), Interne RTL-Dokumentation, v1.0, S. 3
- [StaVaVe 1999] Staudt, M.; Vaduva, A. ;Vetterli, U.: Metadata Management and Data Warehousing In: Software Engineering. Universität Zürich. (Zürich 1999), S. 12
- [Tieto 2000] TietoEnator (o.V.): Output Formats, In: TRIPclassic Manager Guide (Januar 2000), Software Manual, S. 163
- [YoKiYoKiLe 2006] Yoo, S.; Kim, Y.; Youk, S.; Kim, J.; Lee, B.: Adaptive-Binning Color Histogram for Image Information Retrieval In: International Journal of Multimedia and Ubiquitous Engineering, Vol 1. No. 4 (Daejon, Korea 2004).

Internetadressen

- [CBC 2007] Cologne Broadcasting Center GmbH
<http://www.cbc.de/cbc3/index.php?id=41>,
Abruf am 01-06-2007
- [CREATION 2007] RTL Creation GmbH
http://www.rtl-television.de/index_creation.html,
Abruf am 20-06-2007
- [ImageJ 2007] ImageJ
<http://rsb.info.nih.gov/ij/docs/intro.html>
Abruf am 10-07-2007

- [Java 2007] Java
 <http://www.java.com>
 Abruf am 10-07-2007
- [MicroVisual 2007] Microsoft Visual Studio 2005
 <http://msdn2.microsoft.com/de-de/vstudio/default.aspx>,
 Abruf am 29-06-2007
- [MicroSource 2007] Microsoft Visual Source Safe 6.0
 <http://msdn2.microsoft.com/en-us/vstudio/aa700907.aspx>, Abruf am 29-06-2007
- [RTL 2007] RTL Television GmbH
 <http://www.rtl-television.de>, Abruf am 20-06-2007
- [Sapiens 2007] Sapien Technologies - Primalscript
 <http://www.primalscript.com/>, Abruf am 29-06-2007
- [Selfhtml 2007] Selfhtml - HTML
 <http://de.selfhtml.org/intro/technologien/html.htm>,
 Abruf am 21-06-2007
- [SelfhtmlJ 2007] Selfhtml - Javascript
 <http://de.selfhtml.org/intro/technologien/javascript.htm>
 Abruf am 21-06-2007

Danke!

Danke, dass gilt denjenigen, die durch ihre fachliche und persönliche Unterstützung zum Gelingen der Bachelorarbeit beigetragen haben.

Allen voran meiner Freundin Eva Hense, für die Unterstützung und Aufmunterung.

Ein besonderer Dank gilt Prof. Dr. Wolfgang Konen und Dirk Sarembe für die Betreuung der Bachelorarbeit, sowie dem *TRIP*-Team Zoltan Vladescu und Albrecht Tonhauser für die vielen Informationen und Ratschläge.

Außerdem ein großes Lob an Rüdiger Kupke, Josef Jumpers und Sonja Lange für die organisatorische Unterstützung.

Weiterhin möchte ich mich bei Allen bedanken, die meine Arbeit sachkundig kommentiert und somit zur Verbesserung beigetragen haben und bei meinen Eltern, die mir dieses Studium – und damit diese Arbeit – durch ihre Unterstützung erst ermöglicht haben.

Eidesstattliche Erklärung

„Ich versichere an Eides statt durch meine Unterschrift, dass ich die vorstehende Arbeit selbstständig und ohne fremde Hilfe angefertigt und alle Stellen, die ich wörtlich oder annähernd wörtlich aus Veröffentlichungen entnommen habe, als solche kenntlich gemacht habe, mich auch keiner anderen als der angegebenen Literatur oder sonstiger Hilfsmittel bedient habe. Die Arbeit hat in dieser oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen.“

Köln, 10.August 2007