

# Personalisierte Verwaltung und universeller Zugriff auf multimediale Dokument-Strukturen auf Basis von MPEG-7

Diplomarbeit

am Fachbereich Informatik der Universität Rostock



vorgelegt von

**Michael Drews**

geboren am 04.07.1973

**Betreuer:** Prof. Dr. Andreas Heuer, Universität Rostock

Prof. Dr. Bodo Urban, Universität Rostock, Fraunhofer IGD

Dipl.-Inf Guntram Flach, ZGDV e.V. Rostock

Dipl.-Inf. Thomas Courvoisier, ZGDV e.V. Rostock

Abgabedatum: 13. Mai 2002



## **Zusammenfassung**

Mit dem MPEG-7 Standard steht ein umfangreiches Framework für die Beschreibung audiovisueller Daten zur Verfügung, welches Management, Austausch und Suche multimedialer Objekte in einer standardisierten Form ermöglicht.

Diese Diplomarbeit präsentiert die Konzeption einer personalisierten semantischen Selektion und Filterung multimedialer Informationen auf der Grundlage eines serverseitigen Content Managements unter Einbeziehung MPEG-7 konformer Inhaltsinformationen. Dies ist z.B. für den Streaming-Media-Bereich relevant, dem durch den Einsatz von UMTS auch in mobilen Umgebungen zunehmende Bedeutung zukommt. Darüber hinaus ergeben sich signifikante Erweiterungsmöglichkeiten bzgl. der Content-Mediation (Multichannel Publishing) durch die Einbindung von MPEG-7 Beschreibungen in bekannte RDF-Content-Repository-Ansätze.

Ausgehend von einer entsprechenden Deskribierung wurde ein Prototyp implementiert der die entwickelten Konzepte vereint und ihre Anwendung in einem konkreten Szenario, der personalisierten Suche auf Video-Nachrichtenbeiträgen, demonstriert.

## **Abstract**

The MPEG-7 standard provides a rich Framework for the Description of audio-visual Documents which enables the standardised search, management and exchange of multimedia data.

This thesis presents a conception of a server-site content management for the personalised and semantic selection and filtering of multimedia information based on MPEG-7 conform content descriptions. Such concepts are relevant e.g. in the field of Streaming-Media which becomes also important in mobile environments by the use of UMTS and more powerful devices. By combining the presented concepts and known RDF-Content-Repository-Approaches there are new possibilities in terms of content-mediation (multichannel publishing) of audio-visual data.

Based on given content descriptions a prototype is implemented which combines the presented concepts and demonstrates their application in a personalised online video-news scenario.

## **CR-Klassifikation**

E2: Data Storage Representation

H.2.4 Database Management Systems

H.3.3 Information Search and Retrieval

H.5.1 Multimedia Information Systems

## **Keywords**

MPEG-7, Personalisation, RDF, Metadata, Request Processing, Multimedia Information System



## **Danksagung**

An dieser Stelle möchte ich mich bei den Betreuern und Gutachtern dieser Arbeit für die gegebene Unterstützung und hilfreiche Anregungen bedanken.

Besonders möchte ich Guntram Flach und Thomas Courvoisier für die gute Betreuung während meiner Hiwi-Tätigkeit im ZGDV danken.

Weiterer Dank gilt natürlich meinen Eltern für die vielfältige Unterstützung während des Studiums, und schließlich meiner Freundin Barbara für Motivation und liebe Überraschungen auch aus der Ferne.



# Inhaltsverzeichnis

1	Einleitung .....	1
1.1	Motivation.....	1
1.2	Aufgabenstellung .....	2
1.3	Aufbau der Arbeit .....	3
2	Personalisierungskonzepte .....	5
2.1	Überblick .....	5
2.2	Techniken des Profiling .....	9
2.3	Filterungsmethoden.....	11
2.3.1	Content based filtering.....	11
2.3.2	Collaborative Filtering.....	12
2.4	Konzepte für das Anwendungsszenario.....	14
2.5	Fazit .....	15
3	Daten und Metadaten .....	17
3.1	Definition Metadaten .....	17
3.2	Klassifikation von Metadaten .....	18
3.3	Metadatenstandards.....	19
3.3.1	Dublin Core .....	20
3.3.2	MPEG-7.....	21
3.3.3	RDF: Definition von Metadaten .....	22
3.4	Fazit .....	26
4	MPEG-7 .....	27
4.1	Überblick .....	27
4.2	MDS.....	28
4.2.1	Basic Elements .....	29

---

4.2.2	Content Management & Content Description Tools .....	31
4.2.3	Content Organisation / Navigation and Access .....	32
4.2.4	User Interaction.....	32
4.3	User Interaction Tools – Personalisierung mit MPEG-7 .....	32
4.3.1	User Preferences .....	33
4.3.2	Usage History .....	38
4.4	Ausblick: MPEG-21.....	41
4.5	Fazit.....	44
5	Eigene Lösungen.....	45
5.1	Personalisiertes MPEG-7 Content Management.....	45
5.1.1	Reduzierung des Schemas .....	47
5.1.2	Deskriptormenge Videodaten .....	49
5.1.3	Deskriptormenge Nutzerpräferenzen .....	56
5.1.4	Verwaltung der MPEG-7 Beschreibungen .....	58
5.2	Einbindung in XML-Framework .....	62
5.2.1	XPEA-Architektur .....	63
5.2.2	Erweiterungen.....	65
5.3	Prototypische Implementierung <i>pViReSMo</i> .....	74
5.3.1	Überblick .....	74
5.3.2	Anfragekomponente.....	77
5.3.3	Fazit .....	82
6	Schlussbetrachtungen.....	84
6.1	Zusammenfassung.....	84
6.2	Ausblick.....	85
	Literaturverzeichnis .....	87







# 1 Einleitung

## 1.1 Motivation

Die Nutzung digitaler multimedialer Dokumente hat in den letzten Jahren durch technische Entwicklungen im Bereich der Hardware aber auch durch neue Komprimierungs- und Übertragungsverfahren und den damit erleichterten Möglichkeiten der Speicherung und des Austausches enorm zugenommen. Die digitale Form der Speicherung ermöglicht dabei die Archivierung immer größerer Datenbestände. Damit erhöht sich die zur Verfügung stehende Informationsmenge. Um auf dieser großen Informationsmenge arbeiten zu können, muss sie effizient verwaltet werden. Die Verwaltung großer Datenbestände ist die Aufgabe von Datenbankmanagementsystemen.

Um die große Menge elektronisch zur Verfügung stehender multimedialer Daten zu verwalten, und eine Suche auf ihnen zu ermöglichen, müssen die Inhalte dieser Daten geeignet beschrieben werden. Dies erfolgt über sogenannte Metadaten. Im Gegensatz zu Textdokumenten, bei denen der Inhalt, nämlich der Text, leicht extrahiert werden kann, sind bei multimedialen Dokumenten, wie solchen, die Video und/oder Musik enthalten, zusätzliche Beschreibungen notwendig. Der Inhaltsbegriff ist dabei in einem weiteren Sinne zu sehen. Auch Informationen zum Herstellungsprozess, zur Einordnung des Informationsobjektes in eine bestimmte Klassifikation sowie zur Struktur sind als inhaltliche Merkmale anzusehen, die den Verwaltungsprozess unterstützen können.

Es existiert eine Reihe von Standards zur Definition von Metadaten. Einige sind für die Beschreibung unterschiedlichster Informationsobjekte geeignet, andere sind speziell für bestimmte Dokumentarten einzusetzen. Zur Beschreibung audiovisueller Daten existiert mit dem von der MPEG-Group entwickelten MPEG-7 Standard ein umfangreiches Framework, welches in dieser Arbeit eine Grundlage bildet.

Durch breitbandige Übertragungsverfahren steht die große Vielfalt an multimedialen Informationen auch dem Endanwender zur Verfügung. Die Masse der Informationen macht

einen Ansatz, der zur Informationsgewinnung allen Nutzern gleiche Ergebnisse liefert, nicht tragfähig. Hier können Personalisierungskonzepte eingesetzt werden, um den Nutzer bei der Suche zu unterstützen und ihm die Informationen zu präsentieren, die seinen Wünschen und Interessen entsprechen.

Durch neue Übertragungsverfahren ist die Nutzung multimedialer Informationen auch im mobilen Umfeld heute möglich. Dabei sind jedoch, im Gegensatz zu breitbandigen Internetanwendungen mit leistungsfähigen Geräten, an die noch geringeren Leistungsmerkmale der mobilen Geräte und des Netzes angepasste Versionen der multimedialen Dokumente zur Verfügung zu stellen.

## **1.2 Aufgabenstellung**

Ziel dieser Arbeit ist die Konzeption eines serverseitigen Content Managements für die personalisierte, semantische Selektion und Filterung multimedialer, auf der Basis von MPEG-7 beschriebener Dokumente. Des weiteren sollen Möglichkeiten der Kopplung mit bekannten RDF-Repository-Ansätzen, welche eingebettet in eine XML-Framework-Architektur einen universellen, anwendungs- und geräteunabhängigen Zugriff ermöglichen, untersucht werden.

Die personalisierte Suche soll auf einem Videobestand von Nachrichtensendungen stattfinden. Dabei sollen dem Nutzer Beiträge, also Teile der gesamten Sendung, die seinen Interessen entsprechen präsentiert werden. Nutzerinteressen, die bei der Personalisierung in Betracht gezogen werden sollen, sind: bevorzugte Art des Beitrages, bevorzugter Sender der Sendung und bevorzugter Nachrichtensprecher.

Ausgehend von diesem Anwendungsszenario ergeben sich Anforderungen an die Arbeit. Zunächst sind Konzepte der Personalisierung zu untersuchen und entsprechende Möglichkeiten aufzuzeigen. Des weiteren sind die Möglichkeiten, die der MPEG-7 Standard zur Lösung der Aufgabenstellung bereitstellt, zu untersuchen und geeignet anzuwenden sowie Kopplungsmöglichkeiten mit bekannten Content-Repository-Ansätzen zu finden. Schließlich sollen die gefundenen Lösungsansätze in einer prototypischen Implementierung umgesetzt werden.

## 1.3 Aufbau der Arbeit

Angelehnt an die Anforderungen ergibt sich folgender Aufbau der Arbeit. In Kapitel 2 werden zunächst verschiedene Konzepte zur Personalisierung vorgestellt und auf Nutzungsmöglichkeiten im Anwendungsszenario untersucht.

Kapitel 3 gibt einen Überblick über Metadaten, welche für die Durchführung einer personalisierten Suche auf multimedialen Daten unabdingbar sind. Anschließend wird im Kapitel 4 der Metadatenstandard MPEG-7 vorgestellt, welcher ein umfangreiches Framework zur Beschreibung multimedialer Daten zur Verfügung stellt. Dabei stehen besonders die im Standard gegebenen Möglichkeiten der Beschreibung von Nutzerinteressen im Vordergrund. Durch das Zusammenspiel dieser Beschreibung von Nutzern über ihre auf multimediale Dokumente bezogenen Interessen und der Beschreibung der Dokumente selbst ist die Möglichkeit einer Personalisierung auf Basis von MPEG-7 gegeben.

Im Kapitel 5 werden die Lösungen zur Umsetzung der Aufgabenstellung erläutert. Hier werden zunächst die Grundlagen eines personalisierten Content Managements auf MPEG-7 konformen Inhaltsbeschreibungen multimedialer Dokumente gelegt, bevor die Einbindung in eine gegebene XML-Framework-Architektur vorgenommen wird. Zum Abschluss des Kapitels wird eine prototypische Implementierung, die die entwickelten Lösungen umsetzt, vorgestellt.

Im abschließenden Kapitel 6 werden die Ergebnisse der Arbeit zusammengefasst und ein Ausblick auf weitere verwandte Aufgaben gegeben.



## 2 Personalisierungskonzepte

Personalisierung stellt die Mechanismen und Technologien dar, die notwendig sind, den Zugriff auf Informationen durch den Nutzer auf dessen Bedürfnisse und Wünsche anzupassen. Sie kann nach [RsSa01] als automatische Anpassung von Informationsinhalten, -struktur und -darstellung zugeschnitten auf den individuellen Nutzer definiert werden. Personalisierung ist ein Schlagwort, das in unterschiedlichster Weise und mit unterschiedlichen Schwerpunkten verwendet wird, so dass in diesem Zusammenhang auch von „personalized views of personalization“ [Rie00] gesprochen werden kann. Im folgenden werden zunächst einige dieser Sichtweisen vorgestellt, bevor auf Techniken der Personalisierung eingegangen wird.

### 2.1 Überblick

Mittels Personalisierung sollen dem Nutzer/Kunden Informationen auf seine Interessen oder die Anwendungssituation zugeschnitten präsentiert werden. Die Anwendungssituation wird unter anderem durch die Art des Gerätes, mit dem auf die Informationen zugegriffen wird, bestimmt. Eigenschaften der Endgeräte, wie Rechenleistung und Grafikfähigkeiten bestimmen hierbei die Art und den Umfang der zur Verfügung zu stellenden Information. In mobilen Umgebungen sind Ortsinformationen zu den situationsbezogenen Parametern zu zählen. Der Ort bestimmt hierbei die Relevanz von Informationen. So sind etwa die Kulturtips für den Raum Hamburg für einen Nutzer in Berlin relativ uninteressant. Hier wird also sozusagen der Situation eine personalisierte Sicht auf den Datenbestand gegeben.

Eine interessenbezogene Präsentation ist insbesondere in solchen Umgebungen notwendig, in denen durch eine besonders große Anzahl unterschiedlicher Informationen ein Ansatz, der allen Nutzern gleiche Ergebnisse liefert, nicht anwendbar ist. Die Ergebnismenge wäre zu groß und würde zu viele Informationen enthalten, die für den einzelnen Nutzer wenig bedeutsam oder sogar irrelevant sind. Beispiele für solche Umgebungen sind das Internet, wo die Suche nach Begriffen wie „Personalisation“ mit herkömmlichen Suchmaschinen hunderte Ergebnisse liefert, oder auch Digitale Bibliotheken, in denen durch den Einsatz von

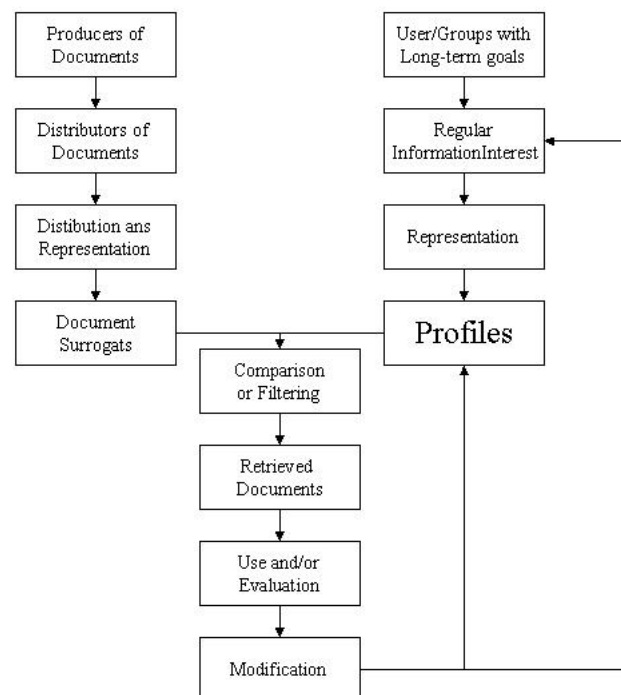
Personalisierungskonzepten gezielter auf neue Informationen zugegriffen werden kann. Personalisierungskonzepte für Digitale Bibliotheken mit dem Schwerpunkt personalisierte Benachrichtigungsdienste werden unter anderem in [Zeit01] vorgestellt.

Durch die gestiegene Rechenleistung vom Heimcomputern sowie große Übertragungsbandbreiten ist die Verarbeitung und Verwaltung von großen Video- und Musikdatenmengen auch für den privaten Nutzer möglich. Hier, aber auch bei der professionellen Video/Musikverwaltung und -recherche, ist Personalisierung für eine effektive Arbeit unerlässlich. Der Einsatz von Personalisierungstechniken ist durch die zunehmende Digitalisierung auch im Bereich des Fernsehens notwendig. Dort kommt es durch die Digitalisierung zu einer immer größeren Programmauswahl. Dies ermöglicht potentiell größeren Fernsehgenuss und eine nie da gewesene Erfüllung von Fernseh Wünschen. Aber auch wenn die technischen Möglichkeiten für diese neue Art des Fernsehens vorhanden sind, so müssen Lösungen für die Handhabung der Programminformationen gefunden werden. Herkömmliche Methoden, relevante Programminformationen zu finden, können wegen der großen Programmfülle nicht mehr eingesetzt werden. So ist etwa Zapping durch alle Programme rein zeitlich nicht mehr möglich und Fernsehzeitungen können nicht alle Programme in ausreichender und handlicher Form darstellen. Auch elektronische Programmführer bieten zur Zeit kaum weniger als eine statische, nach Kategorien geordnete Darstellung des Programms, die nicht auf die speziellen Interessen des einzelnen Nutzers ausgerichtet ist. Hier ist eine Personalisierung des Programmangebotes notwendig, die für den Nutzer relevante Sendungen zusammenstellt.

Personalisierung im Sinne der letztgenannten Beispiele als Auswahl von den Nutzer interessierenden Inhalten aus einem wechselnden Datenbestand wird auch Information Filtering genannt und soll Gegenstand dieser Arbeit sein. Information Filtering ist eng mit konventionellem Information Retrieval verbunden, da beide das Ziel haben Informationen zu liefern, die den Nutzer interessieren und die Zahl irrelevanter Dokumente zu minimieren [FoDu92]. Belkin und Croft stellen in [BeCr92] drei Hauptunterscheidungsmerkmale zwischen Information Retrieval (IR) und Information Filtering (IF) heraus. Die Nutzerprofile des Information Filtering repräsentieren langanhaltende Nutzerinteressen, während die Anfragen in Retrieval System kurzzeitige Interessen widerspiegeln. Zum zweiten werden IF-techniken auf sich ändernden Datenströmen angewandt, während sich beim IR der Zustand der Datenbank nicht oft ändert. Drittens kann Information Filtering als Entfernen von



(irrelevanten) Informationen aus einem Datenstrom und Information Retrieval als Finden von Informationen in diesem Datenstrom angesehen werden. Der letzte Punkt zeigt die Verbindung zwischen IF und IR: um Informationen aus einem Datenstrom herauszufiltern müssen sie zunächst gefunden werden. Hierfür werden beim IF Techniken des IR verwendet. Die Abbildung 2-1 zeigt ein von Belkin und Croft [BeCr92] vorgeschlagenes Modell des Information-Filtering-Prozesses.



**Abbildung 2-1** Filteringprozess nach [BeCr92]

Um dem Nutzer Daten zu präsentieren, die speziell auf seine Bedürfnisse zugeschnitten sind, ist es notwendig, die Interessen (long-term goals) des jeweiligen Nutzers zu ermitteln. Diese können in Angaben zu den Rahmenbedingungen der Suche oder Präferenzen zum eigentlichen Inhalt der gewünschten Suchergebnisse unterteilt werden. Zu ersteren, auch domain- [SmCo99] oder delivery preferences [KBW01] genannt, zählen unter anderem Angaben zur Leistungsfähigkeit des Clientrechners aber auch Angaben zu anderen technischen Voraussetzungen, wie etwa den zu empfangenden Programmen im oben beschriebenen Fernsehzenario. Um die gefundenen Dokumente in einer vom Nutzer

bevorzugten Art zu präsentieren, etwa Layoutwünsche zu berücksichtigen, sind Präferenzen für die Darstellung aufzunehmen.

Inhaltspräferenzen können auf unterschiedliche Arten angegeben werden. Die einfachste Art ist die Angabe in Form von Schlagworten, die gewünschte Eigenschaften darstellen. Die präsentierten Dokumente sollen dann genau diese Eigenschaften besitzen. Die Interessen sind in diesem Fall Bedingungen (conditions), die von der Suche erfüllt werden müssen. Ein erweiterter Ansatz ist die in [LeKi99] und [KBW01] vorgestellte Modellierung von Nutzerinteressen als Präferenzen in teilweiser Ordnung. So kann dann etwa bei der Suche nach Angeboten für ein neues Auto angegeben werden, dass ein blaues Auto mehr bevorzugt wird als ein rotes. Eine Anfrage liefert dann auch Ergebnisse, wenn keine blauen Autos vorhanden sind. Ein solcher Ansatz ist insbesondere im Bereich des e-commerce einzusetzen, wo dem Kunden, anstelle einer leeren Ergebnismenge, Alternativen angeboten werden können, wenn seine „erste Wahl“ nicht erfüllt werden kann. In anderen Anwendungsfeldern sind exakt die Bedingungen erfüllende Ergebnisse gefordert, so etwa bei Informationsdiensten in Digitalen Bibliotheken, bei denen nur Benachrichtigungen über neue Beiträge aus nur einem bestimmten Bereich, wie dem der Datenbanken, gewünscht werden. Die Art der Modellierung von Nutzerinteressen ist also anwendungsabhängig.

Die Nutzerinteressen werden in sogenannten Nutzerprofilen abgelegt. Der Erfolg von Personalisierung hängt von der Verfügbarkeit umfassender und genauer Nutzerprofile ab. Dies bedeutet auch, dass Nutzerprofile an etwaige Änderungen der Nutzerinteressen angepasst werden müssen (Modification). Auch der Inhalt von Nutzerprofilen ist anwendungsspezifisch. Nutzer haben im Nachrichtenbereich andere Interessen als im Bereich der Recherche für Forschung. Es müssen für unterschiedliche Anwendungen also eigene Profile angelegt werden, oder innerhalb des Profils, Kennzeichnungen, die den Anwendungsbereich der einzelnen Interessen angeben, vorgenommen werden.

Zur Filterung von relevanten Dokumenten aus großen Datenbeständen müssen die Dokumente und die sich in den Profilen befindenden Präferenzen in einer Form vorliegen, die beide miteinander vergleichbar macht. Dazu werden die Dokumente indexiert d.h. auf eine Menge von Attributen (Features) die sie kennzeichnen reduziert. Die relevanten Attribute sind sowohl inhaltlicher Natur können aber auch weitere Informationen über das Dokument enthalten. Sie werden auch als Metadaten bezeichnet. Auf das Gebiet der Objektrepräsentation wird in den Kapiteln 3 und 4 näher eingegangen.

Im folgenden werden zunächst mögliche Arten der Erstellung von Nutzerprofilen, sogenannte Profilingtechniken, und anschließend Möglichkeiten der Nutzung von Profilen zur personalisierten Suche, die Filtermethoden, vorgestellt, bevor für das gegebene Anwendungsszenario zu nutzende Personalisierungskonzepte gewählt werden.

## 2.2 Techniken des Profiling

Profiling ist die Sammlung von Präferenzen der Nutzer und deren Verwendung für die Erstellung von Nutzerprofilen. Hierbei können wie oben erwähnt, verschiedene Arten von Präferenzen unterschieden werden. Profilingtechniken werden nach der Beteiligung des Nutzers in zwei Kategorien eingeteilt:

**aktives Profiling:** Nutzer muss Profilinformatoren selbst bereitstellen durch direkte Angabe von Präferenzen oder indirekt über Bewertung von Objekten

**passives Profiling:** Versuch der Ableitung der Nutzerinteressen aus Nutzerverhalten

Als Grundlage für passives Profiling können Daten über Lesezeiten und Browsverhalten (click stream monitoring) oder gezielte Nutzeraktionen, wie der Kauf einer bestimmten Ware in einem Onlineshop dienen. Im Videobereich können neben der Art der gesehenen Sendung auch spezielle Aktionen wie „schneller Vorlauf“, „Zeitlupe“ oder „Standbild“ auf Inhaltspräferenzen des Nutzers hindeuten.

Das Sammeln von Nutzerdaten ist allerdings eine komplexe Aufgabe. Zunächst muss die Bedeutung von Nutzeraktionen festgelegt werden. So kann die Aktion „Standbild“ auf ein vorhandenes Interesse des Nutzers hindeuten. Einerseits in bezug auf die Szene, die er sich genau betrachten will, andererseits auch in bezug auf die gesamte Sendung, die er nach einer Unterbrechung ohne etwas zu verpassen weitersehen möchte.

Die wirkliche Absicht einer Nutzeraktion ist jedoch nicht immer klar feststellbar. So könnte bei Internetanwendungen mehrmals auf einen Link geklickt werden, wenn die Seite nicht schnell genug geladen wird, oder das Lesen eines Textes durch einen Anruf oder ähnliches unterbrochen werden, was im ersten Fall zu einer falsch ermittelten Anzahl von Seitenbesuchen führt und im zweiten die Lesezeit verlängert. Auch eindeutige Aktionen, wie der Kauf eines Buches im Onlineshop, spiegeln nicht in jedem Fall das wirkliche Interesse des Nutzers wider, so zum Beispiel dann, wenn das Buch für einen Freund gekauft wurde.

Zusätzlich zur Bestimmung der Bedeutung von Nutzeraktionen müssen also auch geeignete Methoden gefunden werden, die die Bedeutungen genau erfassen, und Aktionen, die zu einer verzerrten Darstellung der Präferenzen führen, herausfiltern. Hierzu können zum einen Regeln gefunden werden, die solche Aktionen nicht berücksichtigen. So werden beim click stream monitoring in [RaSm01], Klicks, die in einem zu geringen Zeitintervall durchgeführt wurden, nur einmal gezählt, um Mehrfachklicks aus der Präferenzerstellung auszuschließen. Die Lesezeitanalyse kann mit der Analyse von Scrollaktionen kombiniert werden, um Leseпаusen oder das „Überfliegen“ von Passagen zu erkennen. Generell kann mit der Kombination von (passiven) Profilingtechniken eine höhere Genauigkeit des Nutzerprofils erreicht werden. In Onlineshops kann neben dem Kaufverhalten auch das Browsen durch das Angebot und die Art und Weise der Nutzung der durch die Personalisierungskomponente erstellten Vorschläge in die Präferenzenerstellung einbezogen werden.

Beim aktiven Profiling muss der Nutzer die Profilinformationen selbst bereitstellen. Dies geschieht durch die direkte Angabe von Präferenzen bzw. von Inhaltsbewertungen. Die Daten sind zwar, besonders im Vergleich mit den aus passivem Profiling gewonnenen, sehr genau, aber vor allem bei inhaltsbezogenen Präferenzen oft unvollständig. Domänenpräferenzen werden dagegen meist vollständig angegeben [SmCo99]. Um die Qualität der inhaltsbezogenen Präferenzen zu verbessern, können durch den Nutzer explizit angegebenen Bewertungen von Vorschlägen der Personalisierungskomponente in das Profil aufgenommen werden. Diese Art des Profilupdates kann auch zur Qualitätssteigerung der aus dem passiven Profiling gewonnenen Nutzerdaten eingesetzt werden. Sie ist aber auch sinnvoll, um etwaige Veränderungen der Nutzerpräferenzen zu erkennen und in das Profil aufzunehmen.

Bei der Entscheidung, welche Art des Profiling eingesetzt werden soll, ist zu berücksichtigen, inwieweit der Nutzer bereit ist, den Aufwand zur Angabe seiner Interessen zu leisten. Dies hängt von der Applikationsumgebung (wie vertraut ist der Nutzer mit ihr) und dem Wert, den gute Ergebnisse für den Nutzer haben, ab. So zeigt sich bei einer über die Nutzung der Suchmaschine Yahoo! und ihrer rein durch aktives Profiling personalisierten Variante myYahoo! durchgeführten Studie, dass einerseits die Mehrzahl der Nutzer die personalisierte Variante gar nicht nutzen, sondern beim Standard bleiben. Andererseits nutzen diejenigen, die sich eine persönliche Seite angelegt haben, die Personalisierungsmöglichkeiten auch ausgiebig [MPR00]. Es gibt also Anwender, für die eine personalisierte Seite keinen Wert hat, und andere, die größeren Aufwand betreiben, da

sie sich davon einen Nutzen versprechen. In Personalisierungskomponenten sollten damit sowohl aktive Profilingtechniken, die eine höhere Genauigkeit haben, als auch passives Profiling, welches den Nutzer entlastet, eingesetzt werden. Besonders in neuen Anwendungen ist die Akzeptanz für Personalisierung zunächst wohl wenig groß und die Neigung großen Aufwand zu betreiben, gering [Baud98]. Aktive Profilingtechniken sollten aber auch hier nicht außer Acht gelassen werden, da sie auch insbesondere beim Profilupdate klare Vorteile besitzen. Aber auch hier können passive Techniken eingesetzt werden. Besonders Domänenpräferenzen, wie bevorzugte Uhrzeiten und Programme in TV-Umgebungen sind so auf einem aktuellen Stand zu halten.

## 2.3 Filterungsmethoden

Um dem Nutzer auf seine Bedürfnisse und Interessen zugeschnittene Informationen zu präsentieren, werden die in den Nutzerprofilen abgelegten Präferenzen genutzt. Der Datenbestand wird so gefiltert und ggf. in der Darstellung angepasst, dass er den Nutzervorgaben entspricht. Um Datenbestand und Nutzerprofile vergleichen zu können müssen beide als Menge kennzeichnender Attribute (Features) beschrieben werden. Es werden nun solche Dokumente gesucht, bei denen die Werte der kennzeichnenden Attribute mit denen der Präferenzen im Nutzerprofil übereinstimmen. Methoden, die zur Filterung des relevanten Inhaltes eingesetzt werden, werden in content-based filtering, auch cognitive filtering genannt, und collaborative filtering, auch als social filtering bezeichnet, unterteilt. Diese beiden Ansätze werden im folgenden vorgestellt.

### 2.3.1 Content based filtering

Der Ansatz des content based filtering ist, nach Objekten zu suchen die ähnlich denen, die der Nutzer in der Vergangenheit als positiv bewertet hat und verschieden zu denen, die er negativ bewertet hat, sind. Grundlage des Filtering ist hier also der Inhalt von Objekten. Dabei kann Inhalt im weiteren Sinne gesehen werden und sogenannte Metainformationen, wie Autor und Verlag im Bereich von Publikationen, mit einbeziehen. Die Attribute, die im Profil abgelegt sind und die Objekte des Datenbestandes kennzeichnen sind aus dem Inhalt der Objekte abgeleitet. Im Profil sind inhaltsbasierte Features von Objekten abgelegt, die die Präferenzen des Anwenders bzgl. des jeweiligen Features widerspiegeln. Zu beachten ist

hierbei, dass diese Features beim passiven Profiling auf Basis des „Objektkonsums“ des Nutzers aus den Objekten extrahiert bzw. mit ihnen mitgeliefert werden müssen. Um die Attributwerte aus den Nutzerprofilen mit denen die Objekte repräsentierenden vergleichen zu können müssen zufriedenstellende Ähnlichkeitsmodelle gefunden werden. Das einfachste ist sicher das Gleichheitsmodell, bei dem zwei Attribute nur ähnlich sind, wenn sie den gleichen Wert haben. Für weitergehende Ähnlichkeitsbestimmungen sind geeignete Ontologien zu entwickeln, welche die Ähnlichkeiten definieren.

Der inhaltsbasierte Filteransatz hat seine Ursprünge im Information Retrieval Umfeld und verwendet viele Techniken aus diesem Bereich [Baud01]. Dabei wird zwischen dem „exact-match“ und dem „partial-match“ (auch „best-match“) Ansatz unterschieden. Beim exact-match Retrieval werden nur Dokumente zurückgeliefert, die die Anfrage genau erfüllen. Im Sinne einer höheren Effektivität von Anfragen wäre es aber auch wünschenswert, wenn Resultate geliefert würden, die die Anfrage nicht vollständig erfüllen. Mit exact-match Anfragen werden Dokumente in solche eingeteilt, die die Anfrage erfüllen, und solche die dies nicht tun. Eine weitere Unterscheidung kann nicht vorgenommen werden. Insbesondere ist es nicht möglich, die Dokumente in irgendeiner Form nach ihrer Wichtigkeit zu ordnen (relevance ranking) und so dem Anwender die relevantesten Dokumente an vorderster Stelle der Ergebnisliste zu präsentieren, was die Suche komfortabler gestaltet. Eine Ordnung nach der Wichtigkeit der Dokumente ist besonders auch bei großen Ergebnismenge wünschenswert, da die Größe reduziert werden kann, ohne dass die wichtigsten Ergebnisse verloren gehen.

Um diese Einschränkungen aufzuheben, wurden die partial-match Techniken entwickelt. Die bekannteste ist das Vektorraummodell. Dabei werden Objekte und Anfragen als Vektoren im mehrdimensionalen Raum angesehen und Anfrage und Objekt verglichen, indem ihre Vektoren verglichen werden. Eine wesentliche Verbesserung des Modells gegenüber exact-match Anfragen stellt die Möglichkeit dar, dass die Terme der Anfrage oder die Repräsentationen der Dokumente gewichtet werden können.

### **2.3.2 Collaborative Filtering**

Inhaltsbasierende Filtertechniken setzen einzig auf Inhaltsangaben, um für den Nutzer wertvolle Informationen zu ermitteln. Sie haben daher Grenzen, wo solche Angaben schwer zu extrahieren sind. Außerdem sind inhaltliche Angaben nur ein Kriterium bei der

Bewertung eines Objektes. Auch subjektive, schwer zu formalisierend Aspekte spielen eine Rolle. So kann man nach einem Kinobesuch oft nicht genau sagen, warum der Film gefallen hat. Und manchmal kann auch der Lieblingsschauspieler einen schlechten Film nicht retten.

Collaborative Filtertechniken haben nicht den Inhalt von Dokumenten, sondern Bewertungen von Objekten als Grundlage. Während also beim content-based-filtering zwei Dokumente ähnlich sind, wenn ihr Inhalt in den gleichen Features übereinstimmt, so sind sie beim collaborative filtering ähnlich, wenn sie gleich bewertet wurden. Durch diesen Ansatz sind collaborative Filtertechniken auch dann einsetzbar, wenn keine Beschreibungen der Objekte vorliegen. Die Bewertungen der Objekte können wie oben beschrieben durch aktives oder passives Profiling erhalten werden, wobei es hier im Gegensatz zum content based filtering nicht notwendig ist, Inhaltsangaben mitzuliefern oder aus den Objekten zu extrahieren.

Beim collaborative filtering werden den Nutzer Objekte geliefert die ähnliche Nutzer positiv bewertet haben. Die Auswahl dieser Nutzer kann aktiv oder passiv erfolgen. Bei aktiver Wahl kennt der Nutzer andere Nutzer bzw. deren Geschmack und will ihre Bewertungen in die Vorschläge für relevante Objekte einbeziehen.

Passives collaborative filtering, oder automated collaborative filtering, sucht aus allen Nutzern ähnliche automatisch heraus, indem die Bewertungen der Nutzer verglichen werden. Ist eine gewisse Ähnlichkeit vorhanden, so werden Objekte die von diesen ähnlichen Nutzer als positiv bewertet wurden, und die nicht im Nutzerprofil des Suchenden vorhanden sind vorgeschlagen. Die Ermittlung von ähnlichen Nutzern kann in regelmäßigen Abständen erfolgen, was die Antwortzeit bei tatsächlichen Anfragen verkürzt. Ein Vorteil dieser passiven Filtervariante gegenüber der aktiven ist die größere Zahl der Nutzer die in den Vergleich einbezogen werden können, was es ermöglicht, mehr ähnlich Nutzer zu finden. Ein Nachteil ist, dass die Nutzerdaten zentral gehalten werden müssen, was unter Datenschutzaspekten kritisch sein kann. Es müssen zunächst jedoch ausreichend viele Nutzer vorhanden sein, um Gruppen ähnlicher Nutzer zusammenzustellen. Gruppen ähnlicher Nutzer können auch losgelöst vom konkreten Nutzerverhalten und Inhaltsbewertungen ermittelt werden. Dabei werden Nutzer mit ähnlichen Merkmalen in Gruppen zusammengefasst. Mögliche Merkmale sind unter anderem Alter, Geschlecht, Beruf. Bei der Sammlung solcher Daten sind Aspekte des Datenschutzes in Betracht zu ziehen.

Ein generelles Problem kollaborativer Techniken ist, dass nur Objekte vorgeschlagen werden können, die auch schon bewertet worden sind. Besonders neue Objekte sind also nicht unter

den Vorschlägen zu finden. Um dies zu verhindern müssen die Objekte frühzeitig bewertet werden. In [Bau01] werden spezielle Nutzer, sogenannte Opinionleader vorgeschlagen, die neue Objekte, ähnlich einem Redakteur in herkömmlichen Medien, frühzeitig bewerten. Diese Opinionleader können auch in das aktive Filtern einbezogen werden.

Ein anderer Ansatz zur Lösung dieses sogenannten Cold-Start-Problems ist der Einsatz automatischer Bewerter, sogenannter hypothetischer Nutzer, welche Objekte „mögen“, die bestimmte Eigenschaften haben. Umgesetzt wurde diese Konzept beispielsweise in [KRBH98] mit sogenannten „filterbots“ und in [Gree98] mit „Archetypes“. Bei diesem Ansatz ist jedoch wieder eine Ableitung der Qualität eines Objektes aus seinen Eigenschaften notwendig und damit der Vorteil des ohne Inhaltsbeschreibungen auskommenden Ansatzes wieder aufgegeben. Hier werden Techniken der kollaborativen und der inhaltsbasierten Filterung kombiniert.

Neben dem Einsatz zur Lösung des Cold-Start-Problems kann eine Kombination der Filtermethoden auch zur Erweiterung der Suchanfragen genutzt werden. So kann dann nach Dokumenten ähnlicher oder bevorzugter Nutzer und/oder nach Dokumenten mit bestimmten Eigenschaften gesucht werden [Baud 99]. Hier werden sowohl inhaltliche Präferenzen als auch bevorzugte Nutzer in das Profil aufgenommen.

## 2.4 Konzepte für das Anwendungsszenario

Die bisherigen Ausführungen verdeutlichen, dass die einzelnen Personalisierungskonzepte Vor- und Nachteile aufweisen und ihr Einsatz anwendungsspezifisch erfolgen muss. Für den Einsatz in dem im Kapitel 1 vorgestellten Anwendungsszenario können folgende Konzepte genutzt werden.

Bei der Beschreibung der Nutzerinteressen sollten sowohl Präferenzen als auch Bedingungen unterstützt werden können. So kann eine Gewichtung bei breitem Nachrichteninteresse und Einschränkung auf ein spezielles Gebiet bei konkreten Interessen modelliert werden.

Die Filterung der relevanten Beiträge sollte content-based erfolgen. Collaborative Techniken sind im Nachrichtenbereich nicht geeignet, da die Beiträge neu sind und in der Regel keine Bewertung der Beiträge vorgenommen wird. Die für das content based filtering notwendige Beschreibung des Inhaltes erfolgt über Metadaten. Für audiovisuelle Daten existiert mit MPEG-7 ein Beschreibungsstandard, der hier genutzt werden kann.



Zur Ermittlung der Nutzerinteressen wird zunächst aktives Profiling vorgeschlagen. Die Domäne Nachrichtensendung ist dem Nutzer bekannt und anhand von Nachrichtengenre gut strukturiert, so dass eine aktive Angabe der Interessen unkompliziert erscheint. Die Interessen können sich im Nachrichtenbereich aufgrund aktueller Ereignisse rasch ändern, was ein passives Profiling komplexer gestaltet und in der konkreten Umsetzung zu beachten ist.

## 2.5 Fazit

In diesem Kapitel wurde zunächst ein Überblick über den Begriff und die Einsatzgebiete von Personalisierung gegeben. In dieser Arbeit soll Personalisierung als Filterung relevanter Informationen gesehen werden.

Personalisierungskonzepte konnten in der Art der Modellierung von Nutzerinteressen, der Art der Gewinnung der Nutzerinteressen und der Filtermethode unterschieden werden. Jedes der vorgestellten Konzepte hatte Vor- und Nachteile aufzuweisen. Die Wahl der einzusetzenden Konzepte ist von der konkreten Anwendung abhängig.

Für das in dieser Arbeit gewählte Anwendungsszenario wurden passende Personalisierungskonzepte gewählt. Das gewählte content-based-filtering erfordert die Beschreibung von Eigenschaften der audiovisuellen Daten. Dies geschieht über Metadaten. Im folgenden Kapitel wird nun das Konzept der Metadaten vorgestellt.



# 3 Daten und Metadaten

Für die Unterstützung der Suche nach Dokumenten in großen Datenmengen, ist es notwendig, dass die Dokumente in einer Form dargestellt werden, welche sie mit der Anfrage vergleichbar macht und zusätzlich eine effiziente Suche ermöglicht. Diese Aufgabe erfüllen sogenannte Metadaten. Metadaten können auch zur Steuerung von Prozessen eingesetzt werden. In dieser Arbeit wird eine metadatengesteuerte Anfragebearbeitung vorgestellt. Im folgenden wird zunächst eine Definition für Metadaten gegeben bevor unterschiedliche Einteilungsmöglichkeiten (Klassifikationen) von Metadaten vorgestellt werden. Anschließend wird auf einige wichtige Metadatenstandards eingegangen.

## 3.1 Definition Metadaten

Für den Begriff “Metadaten “ existiert keine allgemein anerkannte Definition. Er wird auf vielfältige Weise genutzt und kann so, auf verschiedene Arten betrachtet werden.

Als umfassende Betrachtungsweise sieht Gilliland-Swetland in [Gil00] Metadaten als die „Summe all dessen, was man über ein Informationsobjekt sagen kann“. Metadaten sind in der allgemeinsten Betrachtungsweise also „Daten über Daten“, wobei die beschriebenen Daten als Informationsobjekte bezeichnet werden. Ein Informationsobjekt ist hierbei alles, was durch eine Maschine oder einen Menschen identifiziert und manipuliert werden kann. Alle Informationsobjekte besitzen drei grundlegende Eigenschaften, die durch Metadaten beschrieben werden können. Diese sind [Gill00]:

- **Inhalt:** bezieht sich auf die im Objekt enthaltenen Informationen
- **Kontext:** stellt Aspekte die Objekterzeugung betreffend dar und beantwortet in diesem Zusammenhang Fragen nach dem Wer?, Was?, Warum? und Wo?
- **Struktur :** beschreibt Zusammenhänge innerhalb eines oder zwischen mehreren Informationsobjekten

Am Beispiele des Informationsobjekte Beitrag einer Nachrichtensendung können als inhaltsbezogene Metadaten Ort und Art des Ereignisses über das berichtet wird, als

kontextbezogene Metadaten der für den Beitrag verantwortliche Redakteur oder der Sprecher der Nachricht und als strukturbeschreibende Metadaten die Einteilung in Anmoderation des Berichtes und eigentlicher Bericht angesehen werden.

## 3.2 Klassifikation von Metadaten

Metadaten werden in unterschiedlichsten Bereichen der Informationsverarbeitung eingesetzt. So deckt das Konzept der Metadaten ein breites Spektrum von Nutzungsarten ab. Zum besseren Verständnis der Konzepte hinter den Metadaten ist eine Kategorisierung nach ihrem Einsatzgebieten sinnvoll. Mögliche Gebiete und Beispiele zugehöriger durch Metadaten beschriebene Informationen sind [Gil00]:

- Administration und Verwaltung
- Zugriffsinformationen, Rechte, Versionskontrolle, Ortsinformationen
- Beschreibung und Identifikation
- Katalogeinträge, spezialisierte Indizes, Beziehungen zwischen Informationsobjekten, Nutzerannotationen
- Konservierung
- Dokumentation des physischen Zustandes, Erhaltungsmaßnahmen
- technische Informationen
- Hardware und Softwaredokumentation, Infos zum digitalen Zustand(Auflösung, Bitrate)
- Nutzung
- Nutzungsvorschriften, Nutzertracking, Informationen zur Weiterbenutzung

Neben dieser Kategorisierung nach Einsatzgebieten lassen sich Metadaten nach ihren Eigenschaften kategorisieren. Folgende Kriterien können dabei berücksichtigt werden [Gill00]:

### Quelle der Metadaten

Unterscheidung zwischen internen und externen Metadaten. Interne Metadaten werden bei Erzeugung des zu beschreibenden Objektes durch den Erzeuger oder auch automatisch angelegt. Externe Metadaten werden zu einem späteren Zeitpunkt hinzugefügt. Beispiele für

externe Metadaten sind Daten die zur Katalogisierung der Informationsobjekte dienen, sowie rechtliche Informationen. Zu den internen Metadaten zählt unter anderem das Dateiformat

### **Struktur der Metadaten**

Unterscheidung zwischen einem vorgegebenen Schema folgenden (strukturierten) Metadaten und Metadaten, die keinem vorgegebenen Aufbau entsprechen (unstrukturiert). Zu den strukturierten Metadaten zählen Bibliotheksmetadaten. Unstrukturierte Metadaten sind z.B. freie Textannotationen.

### **Erzeugungsmethode**

Unterscheidung zwischen automatisch generierten Metadaten, wie Schlagwortindizes bei Texten oder Farbhistogrammen in Bildern, und manuell erstellten Metadaten wie Informationen über den Herstellungsprozess eines Dokumentes.

### **Veränderlichkeit der Metadaten**

Unterscheidung zwischen statischen und dynamischen Metadaten. Der Zustand statischer Metadaten ändert sich nach der Erzeugung nicht. Beispiele für solche Metadaten sind Titel und Autor eines Informationsobjektes. Dynamische Metadaten können ihren Zustand ändern, wenn das Informationsobjekt genutzt oder manipuliert wird. Ein Beispiel hierfür ist eine sich ändernde Verzeichnisstruktur.

## **3.3 Metadatenstandards**

Die Effektivität der Suche auf großen Datenbeständen kann durch den Einsatz von die Daten beschreibenden Metadaten erhöht werden. So können zur Unterstützung der Suche auf Textdokumenten neben Indexen, die den Inhalt der Texte repräsentieren, zusätzliche Metadaten eingesetzt werden, die dem Text eine bestimmte Kategorie, etwa die Kategorie Datenbanken, den Namen des Autors oder auch eine kurze Zusammenfassung zuordnen. Dies ermöglicht eine Suche, die nicht nur Ergebnisse liefert, die sich direkt aus der Textrepräsentation der Daten extrahieren lassen. Bei audiovisuellen Daten eignen sich die

direkt aus den Daten extrahierten Metadaten, wie Farbverläufe, nur bedingt zur inhaltsbasierten Suche. Hier ist die Angabe zusätzlicher Metadaten notwendig.

Um Metadaten universell einsetzen zu können, d.h. insbesondere den Austausch von Metadaten zu ermöglichen, werden Standards genutzt. Hier werden den einzelnen zu beschreibenden Eigenschaften Namen und Bedeutungen zugeordnet, so dass eine eindeutige Repräsentation der Metadaten gewährleistet ist. Alle sich an den Standard haltenden Nutzergruppen können die so erstellten Metadatenätze untereinander austauschen und weiterverwenden. Für eine rechnergestützte Verarbeitung von Metadaten müssen diese so beschrieben werden, dass sie automatisch interpretiert werden können, d.h. sie müssen in einer maschinenverständlichen Form vorliegen. Zur Repräsentation von Metadaten in einer maschinenverständlichen Form wurde vom World Wide Web-Consortium (W3C) das Resource Description Framework (RDF) entwickelt, welches standardisierte Elemente und Attribute bereitstellt, die genutzt werden können, um die Metadaten zu beschreiben und ihre Semantik auszudrücken.

Im folgenden werden zunächst die Metadatenstandards Dublin Core und MPEG-7 vorgestellt, bevor auf die Konzepte von RDF näher eingegangen wird.

### 3.3.1 Dublin Core

Ein Hauptanwendungsgebiet für die metadatengestützte Verwaltung von Dokumenten sind traditionell Bibliotheken. Sie verwalten sehr große Datenbestände gedruckten Materials auf denen eine adäquate Suche möglich sein muss. Aus dem Umfeld der (digitalen) Bibliotheken kamen dann auch erste Bestrebungen eine einfache und doch vollständige Metadatenmenge zur Beschreibung elektronischer Dokumente zu finden. Diese mündeten in der Verabschiedung des Dublin Core Standards. Der Standard bestand anfangs aus 13 Elementen zur Beschreibung elektronischer Dokumente. Diese Menge wurde leicht geändert, so dass der Dublin Core letztendlich aus 15 Elementen besteht. Diese sind im einzelnen:

- Titel: Name des Objektes
- Creator: Name der Person oder Personengruppe, die den Inhalt produzierte
- Subject: das im Dokument behandelte Thema
- Description: textuelle Beschreibung des Inhaltes des Dokumentes (z.B. eine Zusammenfassung)
- Publisher: Person oder Organisation, die Inhalt zur Verfügung gestellt hat
- Contributors: Personen, die wesentliche Zuarbeit geleistet haben
- Date: Datum der Publizierung
- Resource Type: Genrebezeichnung

Format: Beschreibung der physikalische Gestalt des Dokuments  
Ressource Identifier: Zeichenkette oder Nummer, die das Objekt eindeutig identifiziert  
Relation: Beziehung zu anderen Objekten  
Source: Objekt von denen das aktuelle Dokument abgeleitet wurde  
Language: Sprachzuordnung für den Inhalt  
Coverage: örtliche und zeitliche Räume, die für das Dokument charakteristisch sind  
Rights: Verweis zur Copyright und anderen Rechten

Die Beschreibung eines Dokumentes auf Basis von Dublin Core kann direkt durch den Autor des Dokumentes oder von Mitarbeitern einer Bibliothek angegeben werden. Sie bedarf keines Expertenwissens.

Die einfachen Metadatenelmente des Dublin Core sind nach Bedarf erweiterbar. Viele andere Metadatenstandards bauen auf Dublin Core auf oder erlauben eine Abbildung auf seine Kernelemente. In [HuLa98] wird eine Möglichkeit präsentiert, Videoinhalte durch eine Erweiterung der 15 Dublin Core Elemente zu beschreiben. Dabei dienen die 15 Basiselemente zur Beschreibung bibliographischer Informationen über das gesamte Dokument. Dies ermöglicht eine kein Expertenwissen voraussetzende interdisziplinäre Suche unabhängig vom Medientyp. Erweiterung spezieller Dublin Core Elemente (Type, Description, Relation, Coverage) können auf unteren Strukturebenen des Videos ausgenutzt werden um eine spezielle medienabhängige Expertensuche zu unterstützen. Mit Description.Camera.Angle kann so z.B. die Kameraposition abgefragt werden. Ein Nachteil dieses Ansatzes ist jedoch, dass durch die semantische Verfeinerung des Dublin Core eventuell die semantische Interoperabilität eingeschränkt wird [HuAr00].

### 3.3.2 MPEG-7

Der MPEG-7 Standard (auch Multimedia Content Description Interface) der Motion Picture Experts Group (MPEG) stellt speziell Beschreibungsmöglichkeiten für audiovisuelle Daten zum Einsatz im multimedialen Umfeld zur Verfügung. Auf der Basis von MPEG-7 können also Beschreibungen (Metadaten) für audiovisuelle Dokumente erstellt werden. Die zur Verfügung gestellte Beschreibungsmöglichkeiten sind dabei äußerst umfangreich. Die Kodierung der Dokumente ist im Gegensatz zu den vorhergehenden MPEG-Standards (MPEG-1, -2 und -4) nicht Gegenstand vom MPEG-7.

Da der MPEG-7 Standard eine Grundlage dieser Arbeit bildet, wird auf ihn im nächsten Kapitel näher eingegangen.

### 3.3.3 RDF: Definition von Metadaten

Die eben vorgestellten Metadatenstandards wurden für spezielle Anwendungsumgebungen entwickelt. Mit dem Resource Description Framework (RDF) hat das W3C eine Infrastruktur entwickelt, die das Erstellen, den Austausch und die Wiederverwendung strukturierter Metadaten ermöglicht. RDF schreibt keine domänenspezifische Semantik vor, sondern bietet durch ein einfaches und doch vielseitiges Datenmodell die Möglichkeit zur anwendungsspezifischen Definition von Metadatenelementen. Dieses Datenmodell wird zusammen mit einer XML-basierten syntaktischen Repräsentation im ersten Teil der RDF Spezifikation, in der *RDF Model and Syntax Specification* [LaSw99], definiert. Beschreibungen der Metadaten werden durch RDF-Schema [BrGu00] ermöglicht. Im folgenden werden das RDF-Datenmodell und RDF-Schema näher vorgestellt.

#### RDF-Datenmodell

Grundlage von RDF ist ein Modell in Form eines gerichteten Graphen, durch den Ressourcen mit Hilfe von Eigenschaften (properties) in Form von Aussagen beschrieben werden können. Das RDF-Datenmodell basiert also auf drei Objekttypen:

**Ressourcen (resources):** Sind durch RDF zu beschreibende Dokumente. Ressourcen können alle Objekte sein, die eindeutig durch eine URI [URI01], also eine eindeutige Adresse oder einen eindeutigen Namen, identifiziert werden können.

**Eigenschaften (properties):** Spezielle Aspekte, Charakteristika oder Attribute einer Ressource. Der Wert (value) einer Eigenschaft kann ein Literal oder eine andere Ressource sein. Jede Eigenschaft hat einen bestimmten Typ über den ihr eine Bedeutung, Beschränkung der annehmbaren Werte und durch die Eigenschaft beschreibbare Ressourcen zugewiesen werden können. Dies geschieht mittels RDF-Schema.

**Aussage (statement):** Die Verknüpfung einer Ressource mit einer Eigenschaft und einem Wert. Diese drei Teile werden auch als *Subjekt*, *Prädikat* und *Objekt* bezeichnet.

Die **einfache Aussage** „Die Ressource `www.informatik.uni-rostock.de/~drews` hat den Autor Michael Drews“ beschreibt anhand ihrer Eigenschaft „Autor“ die Homepage des Autors dieser Arbeit. Dabei ist `www.informatik.uni-rostock.de/~drews` das Subjekt, *Autor* das Prädikat und *Michael Drews* das Objekt der Aussage. Diese Aussage lässt sich graphisch wie

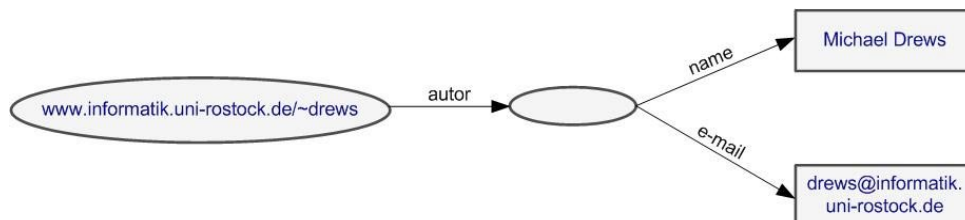


in Abbildung 3-1 gezeigt darstellen. Bei dieser graphischen Notation des RDF-Modells wird eine Ressource durch ein Oval, eine Eigenschaft durch einen ein beschrifteten Pfeil und ein Literal durch ein Rechteck repräsentiert.



**Abbildung 3-1:** einfache RDF-Aussage

Strukturen von Metainformationen sind oft jedoch komplexer als im eben genannten Beispiel. So können Werte von Eigenschaften nicht nur aus einem Literal, sondern aus **strukturierten Werten** bestehen und damit wieder als Ressourcen darstellbar sein. Im Beispiele könnte der Autor der Ressource zusätzlich zum Namen auch noch eine e-mail Adresse besitzen. Die Aussage könnte dann wie folgt lauten: „www.informatik.uni-rostock.de/~drews hat den Autor x und x hat den Namen Michael Drews und die e-mail Adresse drews@informatik.uni-rostock.de“. Die graphische Darstellung dieser Aussage ist der Abbildung 3-2 zu entnehmen.



**Abbildung 3-2:** strukturierte Werte in RDF

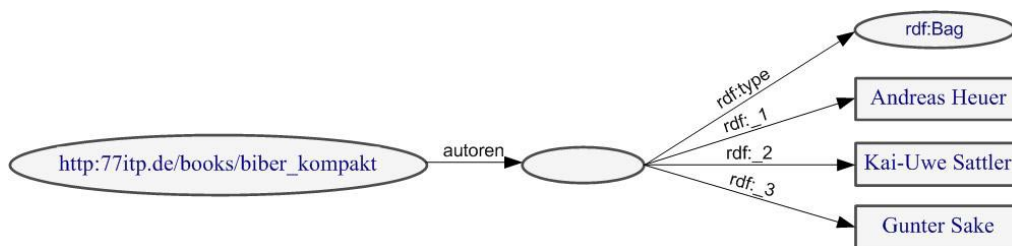
Um einer Eigenschaft mehrere Werte zuzuordnen werden in der RDF-Spezifikation **Container** definiert. Diese können Werte, also Ressourcen oder Literale aufnehmen. Es existieren drei Arten von Containern:

Bag: eine ungeordnete Liste

Sequence: eine geordnete Liste

Alternative: eine Liste, die Alternativen darstellt

Container werden in RDF als Resource mit den Eigenschaften `rdf:type` (Typ des Containers) und `rdf:_1`, ..., `rdf:_n` (die einzelnen Elemente des Containers) angegeben. Ein Beispiel für eine Ressource mit einem Container als Wert einer Eigenschaft ist die Angabe mehrerer Autoren eines Buches also etwa die Aussage: „Die Ressource `http://77itp.de/books/biber_kompakt` hat die Autoren Andreas Heuer, Gunter Sake und Kai-Uwe Sattler“. Die entsprechende graphische Darstellung zeigt Abbildung 3.3.



**Abbildung 3-3:** Container in RDF

Das von W3C vorgestellte RDF-Modell kann in unterschiedlichsten Formen dargestellt werden. Um einen schnellen Überblick über die beschriebenen Ressourcen zu erhalten bietet sich die hier schon genutzte graphische Form an. Für die maschinelle Verarbeitung wird jedoch eine maschinenlesbare Syntax benötigt. Die so erzeugten RDF-Dokumente sollen nicht nur von Maschinen sondern auch von Menschen gelesen werden können und außerdem leicht über Computernetzwerke austauschbar sein. Vom W3C wird hierfür XML als Sprache vorgeschlagen, da sie den genannten Forderungen gerecht wird und außerdem eine hohe Akzeptanz und die Möglichkeit der Erweiterung besitzt. Allerdings sind auch andere Darstellungsformen denkbar. Dabei sind zwei RDF-Ausdrücke, abstrahiert von ihrer Syntax, äquivalent, wenn ihre Modelle identisch sind.

### RDF Schema

Das RDF-Datenmodell bietet die Möglichkeit Ressourcen durch Eigenschaften zu beschreiben, also Metadaten für Ressourcen zu definieren. Aussagen hinsichtlich ihrer Semantik werden jedoch nicht gemacht. Die Metadaten selbst können mittels RDF-Schema (RDFS) beschrieben werden. Darin sind semantische Definitionen für Metadatenelemente und ihre Attribute durch Class- und Propertydeklaration mit Vererbungsmechanismen

sowie durch label- und comment-elemente, durch die die Bedeutung der Elemente beschrieben werden kann, möglich. Mit domain constraint wird die Zugehörigkeit von Properties zu Klassen festgelegt. Mit range-constraint kann angegeben werden, welcher Typ der Wert einer Property haben muss.

Ein Schema wird für eine bestimmte Anwendungsumgebung erstellt und definiert die generellen Anwendungszusammenhänge in dieser Umgebung. Es können auch Metadaten-Schemata definiert werden, die aus Metadatenelementen eines oder mehrerer Metadaten-Schemata abgeleitet und für eine spezielle Applikation optimiert sind. Diese werden auch als Application Profiles [HePa00] bezeichnet. Neue Initiativen, wie TV-Anytime [TVA] und MPEG-21 fordern Application Profiles, die verschiedene existierende Metadaten-Schemata kombinieren wobei Interoperabilität erhalten bleibt und eigene spezifische Forderungen mittels Verfeinerung, Erweiterungen und Ergänzungen erfüllt werden. Application Profiles können außer in RDFS auch in XML-Schema (XMLS) spezifiziert sein.

Um eine Interoperabilität zwischen verschiedenen Application Profiles zu erlangen wird in [HuLA01] eine Web-Metadaten-Architektur vorgeschlagen. Dabei wird die Ansicht vertreten, dass Metadatenspezialisierung bei gleichzeitiger Interoperabilität im Netz am besten erreicht werden kann, wenn jede Metadaten-Domäne sowohl ein RDF-Schema als auch ein XML-Schema für die Domäne in ihrem Namensraum definiert. RDF-Schema dient dabei der Definition von domänenspezifischer Semantik durch Spezifizierung von Typdefinition und -hierarchien. XML-Schema wird zur Spezifizierung der Codierung der Metadatenelemente und Beschreibungen durch Definition von Typen und Elementen sowie deren Content-Modellen, Strukturen, Constraints und Datentypen genutzt, deren Darstellungsmöglichkeiten in RDFS beschränkt sind. Zusätzlich enthalten die XMLSs Links zu den jeweiligen Semantikdefinitionen im RDF-Schema. RDFS wird bei der kombinierten Definition von Application Profiles als übergeordnete Ebene betrachtet.

Durch die Darstellung der einzelnen domänenspezifischen Semantiken in maschinenverständlichem RDF-Schema ist es möglich diese einzelnen Ontologien in einer einzigen Ontologie (auch in RDF-Schema) zusammenzufassen (MetaNet-Ontologie). Damit wäre Interoperabilität zwischen den verschiedenen Application Profiles gegeben und eine Suche über auf unterschiedliche Arten beschriebene Dokumente ermöglicht. Eine Umsetzung dieses Ansatzes, zeigt Hunter in [Hunt01] am Beispiel einer Teilmenge des MPEG-7 Standards, wobei auch Schwierigkeiten bei der Umsetzung der klassenbasierten

XMLS-Defintion auf die Property-zentrierte RDFS Definition aufgezeigt werden und eine Lösung durch die Einbindung der RDFS-Erweiterungen DAML und OIL präsentiert wird.

### **3.4 Fazit**

Dieses Kapitel gab einen Überblick über das Gebiet der Metadaten. Metadaten dienen der Beschreibung von Informationsobjekten und können vielseitig verwendet werden. Das Konzept der Metadaten kann aus verschiedenen Sichtweisen betrachtet werden, so dass es eine Reihe von Klassifikationsmöglichkeiten gibt, von denen einige vorgestellt wurden.

Zum Erstellen von Metadaten stehen Standards zur Verfügung, die den Austausch und die Weiternutzung ermöglichen. Für audiovisuelle Daten existiert der MPEG-7 Standard. Dieser wird im folgenden Kapitel näher vorgestellt.

Mit RDF steht ein Standard zur Beschreibung sämtlicher Objekte, denen ein eindeutiger Name zugeordnet werden kann, zur Verfügung. Dieser wird in der vorliegenden Arbeit zur Beschreibung von Sichten auf Datenbestände genutzt. Die so gewonnenen Metadaten sind dann Grundlage einer universellen Prozesssteuerung.

# 4 MPEG-7

In diesem Kapitel wird der schon erwähnte Metadatenstandard MPEG-7, der eine Grundlage dieser Arbeit bildet, näher vorgestellt. Nach einem Überblick wird besonders auf die sogenannten Multimedia Description Schemes (MDS) eingegangen, die Beschreibungsmöglichkeiten, sowohl für audiovisuelle Daten als auch für Nutzerinteressen, welche diese Daten betreffen, bereitstellen. Anschließend wird ein Ausblick auf den sich in der Entwicklung befindenden MPEG-21 Standard gegeben und die Relevanz für die vorliegende Arbeit untersucht.

## 4.1 Überblick

Der von der Motion Picture Expert Group (MPEG) entwickelte MPEG-7 Standard dient zur *Beschreibung* des Inhaltes audiovisueller Dokumente. Mit ihm können also Metadaten der audiovisuellen Objekte erzeugt werden. Anders als bei vorhergehenden Standards MPEG-1, -2 und -4 ist die Kodierung der Daten nicht Gegenstand von MPEG-7.

Mit MPEG-7 lassen sich audiovisuelle Objekte unabhängig von ihrer Kodierung und der Art ihrer Speicherung beschreiben. So können Beschreibungen von Videos, die auf einem Videoserver gehalten werden, ebenso wie von einem Bild in einer Ausstellung oder von einer selbst komponierten Melodie erstellt werden. Die Art und Weise, wie die Daten aus den zu beschreibenden Objekten extrahiert werden, ist ebenso wenig vorgeschrieben wie die Art der Suche auf den Metadaten. MPEG-7 beschreibt allein die Syntax und Semantik der für die Beschreibung zur Verfügung gestellten Elemente. Grundelemente des MPEG-7 Standards sind:

- **Descriptors (D):** Repräsentationen von Features oder Eigenschaften multimedialer Daten, Syntax und Semantik jeder Feature-Repräsentation ist beschrieben
- **Description schemes (DS):** beschreiben Struktur und Semantik komplexer Features bestehend aus Descriptors und Description Schemes

- **Description Definition Language (DDL):** Sprache zur Definition von Ds und DSs, die auch die Erweiterung und Veränderung erlaubt bestehender Description Schemes erlaubt. Die DDL von MPEG-7 basiert auf XML-Schema.
- **coding schemes:** dienen der effektiven Speicherung (Kompression) von Description Schemes, um effektive Speicherung und Übertragung zu gewährleisten

Der MPEG-7 Standard ist äußerst umfangreich. In ihm sind knapp 300 Beschreibungskomponenten angegeben [Rust01]. Die Spezifikation des Standards ist, entsprechen den MPEG-Gruppen die an ihnen arbeiteten, in sieben verschiedene Teile untergliedert. Diese sind im einzelnen:

- **Systems:** beschreibt Architektur des Standards, Werkzeuge für den effizienten Transport und die Speicherung von MPEG-7 Beschreibungen
- **Description Definition Language:** Definiert Sprache für Beschreibung von Ds und DSs auf Basis von XML-Schema
- **Visual:** Definition von Ds und DSs für visuelle Elemente
- **Audio:** Definition von Ds und DSs für auditive Elemente
- **Multimedia Description Schemes (MDS):** generische Ds und DSs (weder rein visuell noch rein auditiv)
- **Reference Software:** Implementationen relevanter Teile des Standards
- **Conformance:** Richtlinien und Verfahren zum Testen der Konformität von MPEG-7 Implementationen

Im folgenden wird auf Teil 5 des MPEG-7 Standards, die Multimedia Description Schemes [MPEG01a], und dort insbesondere auf die UserInteraction Tools eingegangen. Diese bieten die Möglichkeit der Unterstützung des personalisierten Zugriffs auf multimediale Dokumente. Für einen Überblick über die MPEG-7 Parts Visual und Audio, welche Elemente für die Beschreibung spezieller auditiver und visueller Eigenschaften multimedialer Dokumente, wie Farb- oder Melodieverläufe, zur Verfügung stellen, sei auf [Rust01] verwiesen.

## 4.2 MDS

Das Multimedia Description Scheme (MDS) stellt generische Deskriptoren und Description Schemes zur Beschreibung audiovisueller Dokumente zur Verfügung. Die Einteilung der Ds

und DSs erfolgt hauptsächlich über die Funktionalität, die sie zur Verfügung stellen. Sie sind unterteilt in *basic elements*, *content description*, *content management*, *content organisation*, *navigation and access* und *user interaction*. Die Abbildung 4-1 gibt einen Überblick über die Elemente des MDS, auf die im folgenden näher eingegangen wird.

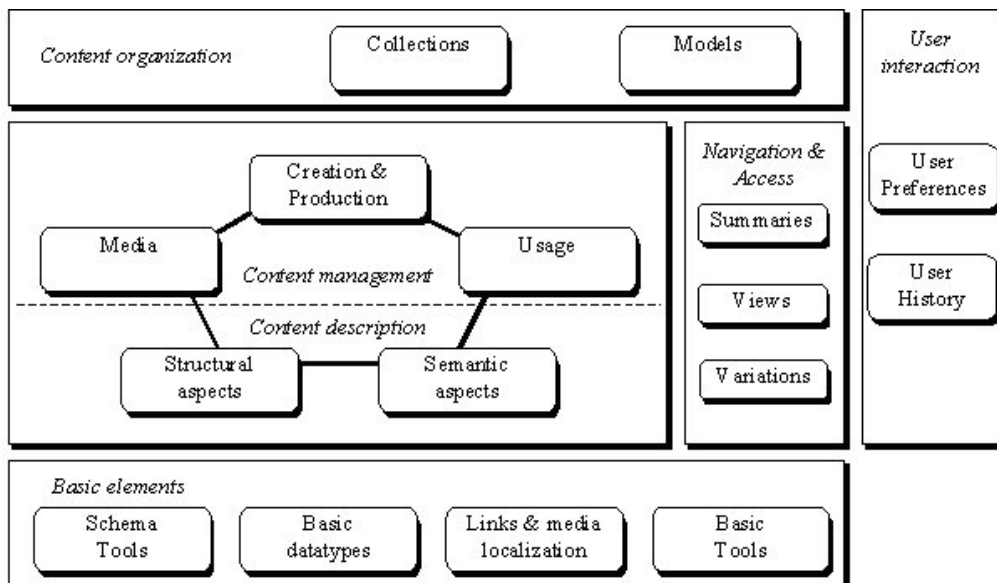


Abbildung 4-1: Elemente des MDS [MPEG01a]

### 4.2.1 Basic Elements

Zu den basic elements zählen schema tools, basic datatypes, Werkzeuge zur Bereitstellung von Links zur Lokalisierung von Inhalten, sowie basic tools, die zur Definition anderer MDS-tools genutzt werden.

Die schema tools definieren das Wurzelement und Top-Level-Elemente von MPEG-7 Beschreibungen, die Top-Level-Datentypen, von denen die anderen MPEG-7 tools abgeleitet sind (z.B. Ds und DSs), sowie packaging tools, die die Zusammenfassung von MPEG-7 Werkzeugen in hierarchisch strukturierten Packages erlauben.

MPEG-7 Beschreibungen beginnen immer mit dem Wurzelement und werden in vollständige (complete) Beschreibungen und Teilbeschreibungen (description units) unterteilt. Dadurch werden sowohl die vollständige als auch die teilweise Übertragung von

Beschreibungen ermöglicht. MPEG-7 Beschreibungen können auch Metadaten, wie Version, Autor und Urheberrechte der Beschreibung, hinzugefügt werden.

Die Top-Level-Elemente folgen in der Beschreibung dem Wurzelement. Zu ihnen zählen multimediale Einheiten, wie Bild, Video, oder Multimediakollektionen, abstrakte Elemente (spezielle Modelle oder Zusammenfassungen von Inhalten) sowie das Management betreffende Elemente (creation und User).

Die basic datatypes sind eingeschränkte integer und real Werte, Vektoren und Matrizen, Länder- und Regionencodes, Typen für die Darstellung von Zeitpunkten und Zeitabschnitten und andere nicht im XML-Schema direkt vorgesehene Datentypen. Beispiele für die mit zu den basic elements zählenden basic tools sind Werkzeuge zur Beschreibung von strukturierten und unstrukturierten Annotationen, Personen, Personengruppen und Organisationen.

Die basic tools ermöglichen außerdem die Definition von Classification Schemes. Ein Classification Scheme (CS) ist eine Menge von festgelegten Termen, die eine Domäne beschreibt. MPEG-7 Beschreibungselementen kann ein solches CS zugeordnet werden, um nur eine vorgeschriebene Menge an Termen zuzulassen und so Mehrdeutigkeiten in der Beschreibung zu verhindern. Terme repräsentieren eine Ausprägung der Domäne, die sie beschreiben. Sie bestehen aus einem Identifier, der den term innerhalb des CS eindeutig identifiziert, einem Namen der dem Nutzer angezeigt werden kann sowie einer Definition, die die Bedeutung des Termes angibt. Die Terme innerhalb eines CS können hierarchisch angeordnet sein, so dass allgemeine und speziellere Terme darstellbar sind. Allgemeine Terme enthalten dabei die speziellen Terme als Kindelemente.

Die Einbindung von Termen aus ClassificationSchemes in MPEG-7 Beschreibungselementen erfolgt über den Datentyp ControlledTermUse Type. Er stellt eine Referenz bestehend aus dem Namen des CS und dem Identifier des Terms, der zur Beschreibung genutzt wird, dar. In zur Beschreibung des Inhaltes genutzten DSs, die nur Terme eines CS enthalten sollen, wird das Attribut *href* vom Typ ControlledTermUse genutzt.



## 4.2.2 Content Management & Content Description Tools

Aufbauend auf den basic elements können Deskriptoren und Description Schemes für die Inhaltsbeschreibungen und Management der Daten genutzt werden. Diese werden mit den Content Management und Content Description Tools zur Verfügung gestellt und beschreiben den Inhalt aus fünf Sichtweisen:

- **Creation & Production** : Beschreibungen, die Herstellung und Produktion betreffend (Titel, Autoren, Mitwirkende, Klassifikation des Inhaltes, eingesetzte Produktionswerkzeuge, ...)
- **Media**: Beschreibung des Medium der audiovisuellen Daten als Menge von Medienprofilen, die unterschiedliche Repräsentationen der Daten bzgl. Qualität, Format u.ä. darstellen
- **Usage**: Metainformationen, die die Nutzung des Inhaltes betreffen (Rechte, Zielpublikum...)
- **Structural Aspects**: Beschreibungen die Struktur des audiovisuellen Inhaltes betreffend (Einteilung in Segmente, die zusätzliche Beschreibungen enthalten können)
- **Semantic**: Beschreibungen des eigentlichen Inhaltes

Die ersten drei Elemente stellen Beschreibungsmöglichkeiten die Verwaltung des Inhaltes betreffend (content management) zur Verfügung. Die anderen beiden dienen hauptsächlich zur Beschreibung von Features, die aus dem Inhalt des audiovisuellen Objektes ableitbar sind (content description). Hierzu zählen die Unterteilung eines audiovisuellen Objektes in zeitliche oder räumliche Regionen (structural aspects) und Beschreibungen der vom audiovisuellen Objekt dargestellten Realität (semantic). Als vertiefende Literatur zu den content description tools sei auf [BZCS01] verwiesen.

Die fünf aufgezählten Elemente sind nicht als getrennte Einheiten zu sehen. Sie stehen untereinander in Beziehung und können teilweise ineinander enthalten sein. So können Media, Usage oder Creation & Production einzelnen Segmenten der strukturellen Beschreibung zugeordnet werden. Dies ist auch in dieser Arbeit umzusetzen. Nachrichtenbeiträge werden als Teil von Nachrichtensendungen beschrieben (zeitliche Segmentierung) und haben ihrerseits eigene zu beschreibende Merkmale, wie etwa einen Titel und einen Sprecher.

### 4.2.3 Content Organisation / Navigation and Access

Neben der direkten Beschreibung multimedialer Elemente können auch Beschreibungen von Sammlungen multimedialer Daten zu einer gemeinsamen Einheit erstellt werden. Hierzu werden Ds und DSs aus den Content Organization Tools genutzt. Dabei können anhand von verschiedenen Modellen auch Beziehungen zwischen solchen Kollektionen beschrieben werden. Mit den *navigation and access tools* können Zusammenfassungen, Partitionierungen und Beziehungen zwischen Variationen multimedialer Daten beschrieben werden.

### 4.2.4 User Interaction

Mit den *user interaction tools* ist die Beschreibung von Nutzerpräferenzen in bezug auf den Konsum multimedialer Daten und von Beschreibungen des zurückliegenden Verhaltens bei der Nutzung multimedialer Daten möglich. Die Beschreibungen sind also nicht, wie die anderen Beschreibungen im MDS an die audiovisuellen Daten gebunden, sondern an Nutzer. Damit werden innerhalb des MPEG-7 Standards Werkzeuge für die Personalisierung multimedialer Datennutzung bereitgestellt. Diese werden im folgenden Abschnitt näher vorgestellt.

## 4.3 User Interaction Tools – Personalisierung mit MPEG-7

Die *user interaction tools* sind in Schemas zur Beschreibung von Präferenzen bei der Nutzung multimedialer Inhalte (User Preferences) und in Schemas zur Beschreibung der Nutzungshistorie multimedialer Inhalte (Usage History) unterteilt.

Die mit User Preferences und Usage History Description verbundenen Grundkonzeption ist in Abbildung 4-2 dargestellt und wird im folgenden erläutert. Hier lassen sich die in Kapitel 2 vorgestellten Personalisierungskonzepte wiederfinden.

Ein Nutzer möchte in einem Multimediasystem audiovisuelle Daten finden. Dem System sind Beschreibungen der multimedialen Inhalte bereitgestellt, so dass eine effiziente Suche, Filterung und Darstellung möglich ist. Die dem System bereitgestellten Beschreibungen der Nutzerpräferenzen werden eingesetzt, um bevorzugte Inhalte zu finden (personalisierte Suche) und/oder um die bevorzugte Form der Präsentation zu wählen (personalisierte Darstellung).

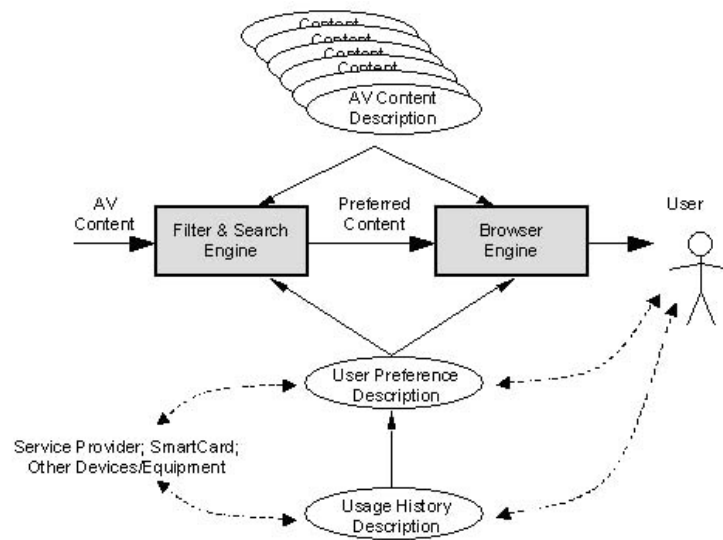


Abbildung 4-2: Personalisierung in MPEG-7 [MPEG01a]

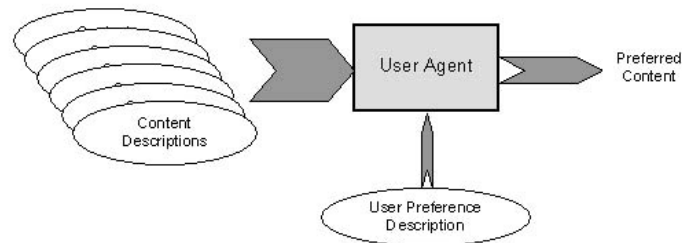
Basierend auf Nutzerinteraktionen mit dem System können Beschreibungen des Nutzerverhaltens generiert werden. Diese Beschreibungen können dann direkt für die personalisierte Filterung eingesetzt oder in Beschreibungen der Nutzerpräferenzen umgewandelt werden (passive profiling). Die Präferenzenbeschreibungen können mit Dritten oder mit anderen Geräten ausgetauscht werden. So sind auch collaborative Filtertechniken anwendbar. Im folgenden werden die einzelnen Teile der UserInteraction Tools analysiert.

### 4.3.1 User Preferences

Die User Preferences tools werden eingesetzt um Nutzerpräferenzen die Verwendung multimedialer Daten betreffend zu beschreiben. Diese Beschreibungen können dann genutzt werden, um auf bevorzugte Inhalte zuzugreifen und diese in bevorzugter Weise darzustellen. Abbildung 4-3 zeigt das von der MPEG in [MPEG01a] vorgestellte generische Modell zur Nutzung von Inhaltsbeschreibungen und Nutzerpräferenzen.

Ein User Agent erhält als Input Inhaltsbeschreibungen und Nutzerpräferenzen und filtert als Ausgabe die Inhalte heraus, die zu den Präferenzen passen. In konkreten Anwendungen kann der Output dann Identifikatoren/Links der bevorzugten multimedialen Inhalte oder auch eine Zusammenfassung des multimedialen Inhaltes, deren Typ den Nutzervorgaben entspricht,

enthalten. Dies entspräche einer Anpassung an Präferenzen den Inhalt bzw. die Darstellung betreffend.



**Abbildung 4-3:** Filtering -Prozess [MPEG01a]

### UserPreferences DS

UserPreferences erlauben die Beschreibung von Nutzerinteressen. Der Name des Nutzers ist innerhalb der UserPreferences im Element `UserIdentifier/UserName` enthalten. Das User Preferences DS ermöglicht die Spezifizierung von Nutzerinteressen für bestimmte Inhalte ebenso wie für bestimmte Arten der Darstellung innerhalb seines Elementes `UsagePreferences`. Hier können im Element `FilteringAndSearchPreferences` Angaben zu bevorzugten Inhalten und in den `BrowsingPreferences` Angaben zur bevorzugten Darstellungsart gemacht werden. Beide Elemente können mehrfach innerhalb eines UsagePreferences-Containers vorkommen.

Zusätzlich kann in den UsagePreferences mit dem Attribut `allowAutomaticUpdate` festgelegt werden, ob es dem System erlaubt ist, die Nutzerpräferenzen automatisch anzupassen. Das Attribut ist vom Typ `userChoice`. Das heißt erlaubte Attributwerte sind: `true` – automatische Änderungen erlaubt -, `false` – automatische Änderungen nicht erlaubt, Änderungen nur manuell durch den Nutzer möglich – und `user` – Nutzer vergibt das Recht auf Änderung fallbezogen auf Anfrage. Mechanismen zur Einhaltung dieser Bedingungen werden im Standard nicht spezifiziert. Dies ist von den jeweiligen Anwendungen zu leisten.

Für diese Arbeit sind besonders die `FilteringAndSearchPreferences` interessant, da in ihnen auf den Inhalt bezogene Interessen beschreibbar sind.

### FilteringAndSearchPreference DS

Mit dem `FilteringAndSearchPreference` DS ist es möglich Nutzerpräferenzen, die Aspekte der Herstellung, der Klassifikation und des Trägermediums des multimedialen Inhaltes betreffen, zu beschreiben. Zur Angabe von Präferenzen aus diesen Bereichen können die Elemente *CreationPreferences*, *ClassificationPreferences* und *SourcePreferences* genutzt werden.

Ein `FilteringAndSearchPreferences` Element kann weitere `FilteringAndSearchPreferences` Elemente enthalten. So sind hierarchische Beziehungen (Spezialisierung) zwischen einzelnen Präferenzen möglich. Die allgemeine Präferenz hat spezielle Präferenzen als Elemente.

Jedes der eben genannten Elemente kann mehrfach innerhalb einer `FilteringAndSearchPreference` instanziiert werden. Die Wertigkeit eines Elementes gegenüber anderen wird über das Attribut *preferenceValue* angegeben. Dieses kann Integerwerte zwischen 100 und -100 annehmen, wobei negative Werte bedeuten, dass Inhalte mit diesen Eigenschaften nicht gewünscht sind. Durch die Möglichkeit der Mehrfachinstanzierung von Nutzerpräferenzen mit unterschiedlicher Gewichtung können in MPEG-7 Präferenzen, wie in Kapitel 2 beschrieben, als Bevorzugung eines Eigenschaftswertes über einen anderen modelliert werden.

Das `FilteringAndSearchPreferences` DS besitzt zusätzlich das Element *preferenceCondition*. Darin kann angegeben werden, unter welchen Bedingungen die Präferenz angewandt werden soll. Hier sind sowohl zeitliche als auch räumliche Bedingungen angebar. So lässt sich aus mehreren, in den `UsagePreferences` angegebene Präferenzen, diejenige auswählen, welche die Bedingung erfüllt.

Zur Filterung relevanter Inhalte können die Werte aus den Präferenzen mit den Werten aus den Beschreibungen der audiovisuellen Inhalte verglichen werden.

#### Creation Preferences

In den `CreationPreferences` sind Präferenzen die Herstellung des Inhaltes betreffend beschreibbar. Hierfür können die Elemente *Title*, *Creator*, *Keyword*, *Location* und *DatePeriode* genutzt werden. Alle Elemente können wieder mehrfach vorkommen und mittels *preferenceValue* gewichtet werden.

Das Element *Title* beschreibt Präferenzen in Bezug auf den Titel Dokumentes. Mit *Keyword* kann ein Schlagwort oder eine Phrase angegeben werden, welche die Inhaltspräferenzen des Nutzers beschreiben.

*Location* beschreibt den bevorzugten Produktionsort und *DatePeriode* eine bevorzugte Zeitspanne, in der der Inhalt produziert wurde.

Mit *Creator* können Präferenzen über Personen angegeben werden, die am Produktionsprozess beteiligt sind. Mit *Role* ist hierbei die Art der Beteiligung (Schauspieler, Regisseur, Sprecher...) angebar. Im folgenden Beispiel sind unterschiedlich gewichtete Präferenzen für zwei Schauspieler angegeben.

```
<FilteringAndSearchPreferences>
  <CreationPreferences>
    <Creator preferenceValue="20">
      <Role href="creatorCS"><Name>Actor</Name></Role>
      <Agent xsi:type="PersonType">
        <Name>
          <GivenName>Brad</GivenName>
          <FamilyName>Pitt</FamilyName>
        </Name>
      </Agent>
    </Creator>
    <Creator preferenceValue="50">
      <Role href="creatorCS"><Name>Actor</Name></Role>
      <Agent xsi:type="PersonType">
        <Name>
          <GivenName>George</GivenName>
          <FamilyName>Clooney</FamilyName>
        </Name>
      </Agent>
    </Creator>
  </CreationPreferences>
</FilteringAndSearchPreferences>
```

#### Beispiel 4-1: gewichtete Präferenzen

#### Classification Preferences

Mit den *ClassificationPreferences* sind Nutzervorlieben die Klassifikation des Inhalts betreffend beschreibbar. Es können bevorzugte Erscheinungsorte und -daten, Sprachen, Genre, Arten, Reviews und Jugendschutzbestimmungen des Inhaltes angegeben werden. Die entsprechenden Elemente sind *Country*, *DatePeriode*, *Language*, *Genre*, *Subject*, *MediaReview* und *ParentalGuidance*.

Herauszustellen ist das Element *MediaReview*, über das bevorzugte Bewerter (Reviewer) multimedialer Inhalte angegeben werden können. Damit können (aktive) collaborative Filtertechniken (vgl. Abschnitt 2.3.2) angewendet werden. Die Elemente der Classification Preferences können, wie auch die einzelnen CreationPreferences-Elemente, mehrfach genutzt und mittels preferenceValue-Attribut gewichtet werden.

### Source Preferences

Über die *SourcePreferences* können Präferenzen angegeben werden, die das Medium selbst oder die Verteilung des Inhaltes betreffen.

Über das Element *PublicationType* kann das bevorzugte Publikationsmedium angegeben werden. Beispiele sind terrestrische Übertragung, Streaming-Übertragung, oder Vertrieb auf CD-ROM.

Mit *PublicationSource* ist eine bestimmte Informationsquelle angebar, etwa ein bestimmter Fernsehkanal oder ein bestimmter Server. *PublicationPlace* bestimmt den Ort von wo aus der Inhalt verteilt wird, *PublicationDate* den Zeitpunkt/Zeitraum, an dem der Inhalt zur Verfügung stehen soll.

Mittels *Publisher* könne Präferenzen, die Person oder Organisation die den Inhalt zur Verfügung stellt, betreffend, angegeben werden. Das Element *MediaFormat* ermöglicht die Angabe von Präferenzen das Format des Mediums des Inhaltes betreffend.

Die Elemente können wieder mehrfach innerhalb einer *SourcePreference* auftreten und mittels *preferenceValue* gewichtet werden. Zusätzlich sind in den *sourcePreferences* optional die Attribute *noRepeat*, *noEncryption* und *noPayPerUse* vom Typ boolean angebar, die anzeigen, ob der Nutzer Inhalte, die schon zugänglich gemacht wurden, verschlüsselt sind bzw. nur gegen Bezahlung zu erhalten sind, bevorzugt.

### Einsatz der *FilteringAndSearchPreferences*

Die *FilteringAndSearchPreferences* stellen umfangreiche Möglichkeiten zur Beschreibung von Nutzerinteressen, die Eigenschaften audiovisueller Daten betreffend bereit. Sie können genutzt werden, um durch den Vergleich der Werte der einzelnen Elemente mit den Werten von den Beschreibungen des multimedialen Inhaltes, relevante Inhalte aus einer großen Informationsmenge herauszufiltern. Es wird also ein content-based filtering unterstützt. Ein Abgleich zwischen Dokument und Interessen kann effizient und genau stattfinden, da die Elemente aus den einzelnen *Preference DSs* direkt oder indirekt eine Entsprechung in den *DSs* zur Inhaltsbeschreibung haben..

So sind bis auf *datePeriode* und *Keyword* sind alle Elemente der *CreationPreferences* direkt auch im *Creation DS* innerhalb der *Creation&Production tools* (siehe oben) vorhanden. Als

Vergleichselement für datePeriode befindet sich im Creation DS das Element CreationDate, welches den Zeitpunkt der Produktion beschrieben werden kann. Das Element Keyword kann mit verschiedenen Elementen aus dem Creation DS verglichen werden, um relevante Dokumente zu finden. Hier bieten sich verschiedene textuelle Komponenten der Inhaltsbeschreibungen wie Title, Abstract oder andere für den Vergleich an. Eine Komponente des UsagePreference DS kann also mit einer oder mehreren aus den Beschreibungen verglichen werden. Allerdings machen natürlich nicht alle Vergleiche Sinn. Verfahren, die angeben welche Elemente mit welchen zu vergleichen sind, sind nicht Teil des Standards. Die Elemente aus den ClassificationPreferences finden ihre Entsprechungen im Classification DS innerhalb der Creation&Production tools. Werte der SourcePreferences finden sich in Instanzen der MediaInformation DS, CreationInformation DS und UsageInformation DS.

Neben den den so gegebenen Möglichkeiten des content based filtering kann mit den FilteringAndSearchPreferences über das Element MediaReview in den ClassificationPreferences auch das collaborative filtering unterstützt werden.

### **BrowsingPreferences**

Die Browsing Preferences können genutzt werden um Nutzerpräferenzen bezüglich der Navigation in und des Zugriffs auf Inhalte zu beschreiben. Hierbei können Präferenzen über die Art und den Inhalt von Zusammenfassungen multimedialer Inhalte angegeben werden. Die BrowsingPreferences beinhalten dazu die Elemente SummaryPreference und PreferenceCondition. Letzteres wurde um das Element Genre erweitert, so dass neben Ort und Zeit auch ein Genre als Bedingung für eine BrowsingPreference angegeben werden kann. Zusätzlich haben BrowsingPreferences die Attribute protected, welches angibt, ob die Informationen an Dritte weitergegeben werden dürfen, und preferenceValue, das wieder eine Gewichtung ermöglicht.

## **4.3.2 Usage History**

Das zweite Werkzeug zur Unterstützung von Nutzerinteraktion im MDS sind die UsageHistory DSs. Diese Schemas können genutzt werden, um Aktionen, die Nutzer auf multimedialen Dokumenten durchführten, zu beschreiben. Abbildung 4-4 zeigt das von



MPEG angegebene Schema zur Ermittlung und Einsatzmöglichkeiten der Usage History Beschreibungen. Hier ist das gleiche Konzept, wie in dem in Kapitel 2 vorgestellten passiven Profiling zu erkennen. Aus Aussagen über das Nutzerverhalten sollen Informationen über Nutzerpräferenzen gewonnen werden.

Die Applikation, mit welcher der Nutzer auf multimediale Inhalte zugreift, erstellt aus den Nutzeraktionen auf multimedialen Dokumenten Beschreibungen des Nutzerverhaltens. Dabei fließen auch Inhaltsinformationen der Dokumente ein. Diese Nutzerinformationen können dann durch einen Profiling Agent in Nutzerprofilbeschreibungen umgewandelt werden.

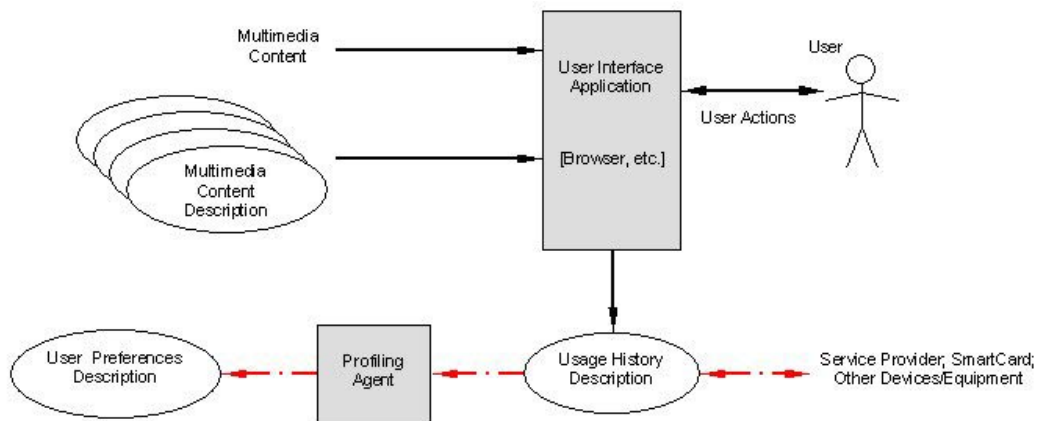


Abbildung 4-4: Ermittlung von Nutzerpräferenzen nach [MPEG01a]

### Usage History DS

Zur Beschreibung des Nutzerverhaltens wird das UsageHistory DS genutzt welches, aus den Elementen *UserIdentifier*, zur Identifizierung des Nutzers, und *UsageActionHistory* mit dem die Nutzeraktionen beschrieben werden, besteht. Zusätzlich wird mit dem Attribut *allowCollection* angegeben, ob der Nutzer die Sammlung seiner Aktionen erlaubt.

### UserActionHistory DS

Das UsageActionHistory DS besteht aus den Elementen *ObservationPeriod*, zur Angabe des Zeitraumes in dem die Nutzerdaten gesammelt wurden, und *UserActionList*, welche die

Aktionen des Nutzers enthält. Außerdem kann über das Attribut *protected* wiederum angegeben werden, ob die Informationen an Dritte weitergeben werden dürfen.

Eine *UserActionList* enthält einen *ActionType* und dazugehörige *UserActions*. Der *ActionType* gibt die Art der Nutzeraktion an. Beispiele wären hier „View“, „Pause“, „Play“ usw. . Die erlaubten Ausdrücke finden sich in *ClassificationSchemes* (siehe oben), wie dem MPEG-7 *ActionTypeCS*. Alle *UserAction* Elemente in einer *UserActionList* haben den gleichen *ActionType*. Es werden also in einer Liste, gleichartige Nutzeraktionen, wie etwa „Record“ mit speziellen Informationen zur einzelnen Aktion gesammelt.

Diese Informationen werden im *UserAction* Element beschrieben. Im einzelnen sind dies: Zeitpunkt und Dauer der Aktion, ID des Programms auf dem die Aktion erfolgte sowie eine Referenz auf Teile der Beschreibungen des Programms oder anderem Related Material. Diese Referenz, im Element *ActionDataItem* beschrieben, sollte auf eine Inhaltsbeschreibung verweisen. So sind dann Ableitungen von Nutzerpräferenzen möglich, die sich auf Beschreibungen der bevorzugten Dokumente beziehen und nicht nur auf die Dokumente selbst.

Zeitpunkt und Dauer der Nutzeraktion sind im Element *ActionTime* sowohl in Bezug auf das Dokument, auf dem die Aktion ausgeführt wurde (innerhalb des Elementes *ActionMediaTime*), als auch als globale Zeitangabe und Dauer (*ActionGeneralTime*) beschreibbar. Die Angaben zur Medienzeit können genutzt werden, um zu erkennen auf welchen Segmenten die Aktion ausgeführt wurde. So kann beschrieben werden, wo etwa ein Video bei Abspielen angehalten worden ist, oder in welchen Bereichen „vorgespult“ wurde. Für letzteres Beispiel ist auch die Angabe der Medienzeitdauer notwendig. Zur Angabe der Dauer der Nutzeraktion selbst wird das Element *ActionGeneralTime* genutzt. Hier ist dann ablesbar wie lange das ein Video angehalten wurde, oder auch wie lange ein Bild oder ein anderes multimediales Dokument „konsumiert“ wurde.

Mit der Möglichkeit der Angabe von Nutzeraktionen, Dauer der Aktionen und Verweisen auf Beschreibungen der Objekte der Aktionen bietet das *UsageHistory* DS einen umfassenden Rahmen zur Beschreibung von Nutzerverhalten. Diese können in den Prozess der Profilbildung durch passives Profiling (vgl.: Abschnitt 2.1) einfließen. Da MPEG-7 ein Beschreibungsstandard ist, sind Methoden des Profiling nicht angegeben. Die Beschreibungen sind aber für derzeit genutzte Methoden ausreichend geeignet.

## 4.4 Ausblick: MPEG-21

MPEG-7 ist ein Beschreibungsstandard und enthält keinerlei Bestimmungen zur Nutzung der Daten. Insbesondere werden keine Angaben zur Einhaltung von Integritätsbedingungen zur Identifikation audiovisueller Daten, zur Durchsetzung des Urheberrechtes bei der Verteilung/Nutzung der multimedialen Informationen, zur Durchsetzung des Datenschutzes bei der Aufzeichnung von Nutzerverhalten sowie allgemein zur Art der einzusetzenden Personalisierungskonzepte gemacht.

Der Bereich der Nutzung audiovisueller und allgemein multimedialer Daten ist Gegenstand des sich in der Entwicklung befindenden MPEG-21 Standards (Multimedia Framework), dessen Eckpunkte im folgenden skizziert werden.

### **MPEG-21 Multimedia Framework**

Die Vision von MPEG-21 ist es, ein Multimedia-Framework zu definieren welches „eine transparente zunehmende Nutzung von Multimediaressourcen über eine große Anzahl von Netzwerken und auf verschiedenen Endgeräten von verschiedenen Organisationen“ ermöglicht [MPEG01d]. Dabei soll die gesamte Verteilungskette multimedialer Inhalte, welche die Entwicklung, Produktion, Verteilung und den Konsum umfasst, unterstützt werden. Dazu müssen die Inhalte identifiziert, beschrieben, verwaltet und geschützt werden können. Die Verteilung der Inhalte erfolgt über ein heterogenes Netzwerk, in dem eine zuverlässige Übertragung, die Verwaltung persönlicher Daten und Präferenzen unter der Einhaltung von Datenschutzbestimmungen sowie die Unterstützung finanzieller Transaktionen gesichert sein muss. Diese Ziele soll MPEG-21 erreichen.

Im Zentrum des MPEG-21 Standards stehen Nutzer multimedialer Inhalte. Diese Inhalte werden hier Digital Item genannt. MPEG-21 stellt einem Rahmen zur Verfügung in dem ein Nutzer mit einem anderen interagiert. Die Nutzer sind durch ihre Beziehung zu anderen Nutzern während einer bestimmten Interaktion identifiziert. Alle Akteure innerhalb von MPEG-21 werden als Nutzer bezeichnet. Es wird kein Unterschied zwischen „Bereitsteller“ und „Konsument“ von multimedialen Inhalten gemacht. Allerdings kann ein Nutzer spezielle Rechte während der Interaktion mit anderen Nutzern erhalten.

MPEG-21 will die Schlüsselemente, die zur Unterstützung der Verteilungskette notwendig sind, sowie die Beziehungen zwischen ihnen und die Operationen, die von ihnen zu

unterstützen sind, identifizieren und definieren. Dabei wird das Multimedia-Framework als eine generische Architektur konzeptionellen Designs beschrieben. Spezielle Anwendungen können auf Grundlage dieser generischen Architektur erarbeitet werden. Eine konkrete Instanzierung und Implementierung der Elemente des Multimediaframeworks unterhalb des für die Interoperabilität notwendigen Abstraktionslevels wird im Standard nicht spezifiziert. Die sieben Schlüsselemente die definiert wurden sind:

**Digital Item Declaration:** einheitliches und flexibles, abstraktes und interoperables Schema für die Definition Digitaler Items (DIs). Ein DI ist dabei jedes digitale Objekt das multimediale Elemente enthält.

**Digital Item Identification and Description :** Framework für die Identifizierung und Beschreibung aller Objekte unabhängig von ihrer Art ihren Typs und ihrer Granularität. Dabei soll die Kombination mit bereits entwickelten Beschreibungsstandards wie MPEG-7 und Dublin Core unterstützt werden.

**Content Handling und Usage:** Bereitstellung von Schnittstellen und Protokollen, welche Erzeugung, Suche, Zugriff, Speicherung, Verteilung und (Wieder-) Verwendung von Inhalten ermöglichen. Besonderer Schwerpunkt soll dabei auf die Verbesserung des Nutzerinteraktionsmodells durch Personalisierung gelegt werden.

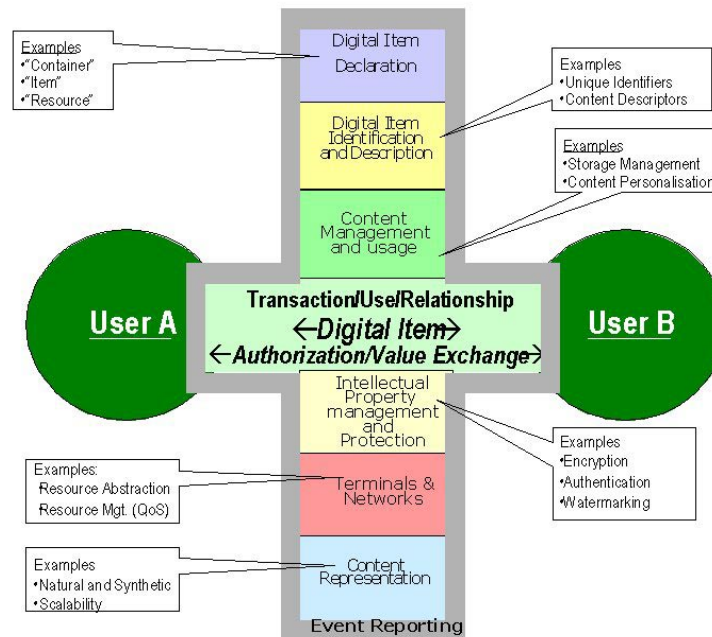
**Intellectual Property Management and Protection:** Framework, dass es allen Nutzern ermöglicht ihre Rechte und Interessen an und Nutzungsbestimmungen für DIs zu bestimmen und eine dauerhafte und Sichere Einhaltung zu gewährleisten

**Terminals and Networks:** Ermöglichung eines interoperablen und transparenten Zugriff auf (verteilte) DIs

**Content Representation:** Bereitstellung und Integration von Repräsentationstechnologien um MPEG-21 Inhalte skalierbar und fehlertoleranten darzustellen

**Event Reporting:** Schnittstellen und Metriken zur Darstellung aller in MPEG-21 auftretenden Events, die sich auf Dis, Prozesse, Transaktionen und Nutzer beziehen

Die Abbildung 4-4 zeigt die Elemente des MPEG-21 Frameworks und das darüber liegende Nutzermodell im Überblick.



**Abbildung 4-5:** Elemente des MPEG-21 Frameworks nach [MPEG01d]

Wie schon erwähnt, befindet sich der MPEG-21 Standard noch in der Entwicklung. Basierend auf den oben beschriebenen Grundelementen hat MPEG-21 einen Work-Plan für die künftige Entwicklung des Standards entwickelt. Darin besteht der Standard aus derzeit aus sieben Teilen:

- Part 1: Vision Technologies and Strategy
- Part 2: Digital Item Declaration
- Part 3: Digital Item Identification and Description
- Part 4 : Intellectual Property Management and Protection (IPMP)
- Part 5: Rights Data Dictionary
- Part 6: Rights Expression Language
- Part 7 Digital Item Usage Environment Description

Die Teile 1 bis 4 befinden sich derzeit auf dem Committee Draft Level, die anderen noch in der Call For Proposals Phase oder in der Vorstufe dazu (Part7). Das für diese Arbeit interessante Schlüsselement Content Handling and Usage findet sich in keinem der Teile des Standards wieder. Es wird zwar daraufhingewiesen, dass dem Standard zusätzliche Teile

hinzugefügt werden, wenn dies erforderlich ist, jedoch scheint es so, dass der Fokus sich mehr auf die Gewährleistung des Schutzes von geistigem Eigentum richtet. Das lässt sich zum einen daraus ableiten, dass zusätzlich zum IPMP-Part die Parts 5 und 6 aufgenommen wurden, die sich mit der Beschreibung von Rechten an DIs befassen. Zum anderen drehen sich die in [MPEG01e] beschriebenen Anwendungsszenarios für MPEG-21 um das Management der Rechte multimediale Inhalte.

## 4.5 Fazit

In diesem Kapitel wurde der MPEG-7 Standard als Beschreibungsstandard für audiovisuelle Daten näher vorgestellt. Neben der Beschreibung der Daten können mit MPEG-7 auch Beschreibungen von Nutzerinteressen und Nutzerverhalten vorgenommen werden, was einen auf standardisierten Nutzerinformationen beruhenden personalisierten Zugriff auf audiovisuelle Daten ermöglicht. Es zeigte sich, dass sich mit diesen sehr umfangreichen Beschreibungsmöglichkeiten die in Kapitel 2 vorgestellten Konzepte zur Personalisierung umsetzen lassen.

Mit dem MPEG-7 Standard steht ein sehr umfangreiches Framework für die Beschreibung audiovisueller Daten zur Verfügung. Für konkrete Anwendungen ist jedoch nicht der gesamte Beschreibungsumfang notwendig. Das Schema sollte auf die notwendigen Descriptoren und Description Schemes reduziert werden. So verringert sich zwar die Flexibilität im Einsatz der Beschreibungen, die Nutzung vereinfacht sich jedoch.

Zur Unterstützung der Auswahl geeigneter Descriptoren ist eine Erweiterung des Standards um die Definition von so genannten MPEG-7 Profiles geplant, in denen für bestimmte Anwendungsfelder zu benutzende Ds und DSs angegeben sind [MPEG01c]. Im folgenden Kapitel wird ein für diese Arbeit genutztes reduziertes Schema vorgestellt.

Auch bei reduziertem Schema ist die Erstellung von MPEG-7 Beschreibungen eine komplexe Aufgabe, die eine Rechnerunterstützung und Expertenwissen erfordert. Eine vollautomatische Deskribierung kann aber nicht in jedem Fall erfolgen, da viele Features, wie etwa Informationen zum Produktionsprozess, nicht direkt aus dem codiert vorliegenden Videomaterial zu extrahieren sind.

# 5 Eigene Lösungen

In den vorangegangenen Kapiteln wurden grundlegende Konzepte für die Durchführung eines personalisierten universellen Content Managements audiovisueller Daten vorgestellt. In diesem Kapitel werden nun in diesem Zusammenhang entwickelten Lösungen vorgestellt.

Dabei werden zunächst die Grundlagen für ein personalisierte Content Management audiovisueller Daten gelegt: Das zur Beschreibung der Daten und der Nutzerinteressen genutzte MPEG-7 Schema wird auf die notwendigen Beschreibungselemente reduziert, bevor eine Abbildung der erhaltenen Schemata auf ein relationales Datenbankschema vorgestellt wird.

Zur Realisierung eines universellen Zugriffs erfolgt eine Integration in eine bestehende XML-Framework-Architektur. Nach einem Überblick über das Framework werden notwendige Erweiterungen vorgestellt.

Die entwickelten Konzepte werden abschließend in einer prototypischen Implementierung umgesetzt.

## 5.1 Personalisiertes MPEG-7 Content Management

Wie in Kapitel 3 gezeigt, liegt ein wesentliches Einsatzgebiet von Metadaten in dem Bereich der Beschreibung des Inhaltes audiovisueller Daten, da aus deren kodierter Datenrepräsentation allein kaum für die Suche nutzbare Inhaltsinformationen ableitbar sind. Als Metadatenstandard für die Beschreibung audiovisueller Daten wird in dieser Arbeit MPEG-7 genutzt. Der Standard wurde im vorhergehenden Kapitel bereits beschrieben. Für konkrete Anwendungen wird nicht die gesamte Beschreibungsmächtigkeit von MPEG-7 benötigt. Es kann eine Menge von Deskriptoren und Discription Schemes ausgewählt werden, welche die in der Anwendung gewünschten Eigenschaften der audiovisuellen Daten adäquat beschreiben. Eigenschaften von Videos können dabei in folgende Gruppen unterteilt werden [Rehm00,Rust01].

**Struktur:** Angabe zu Videounterteilung in einzelne Segmente

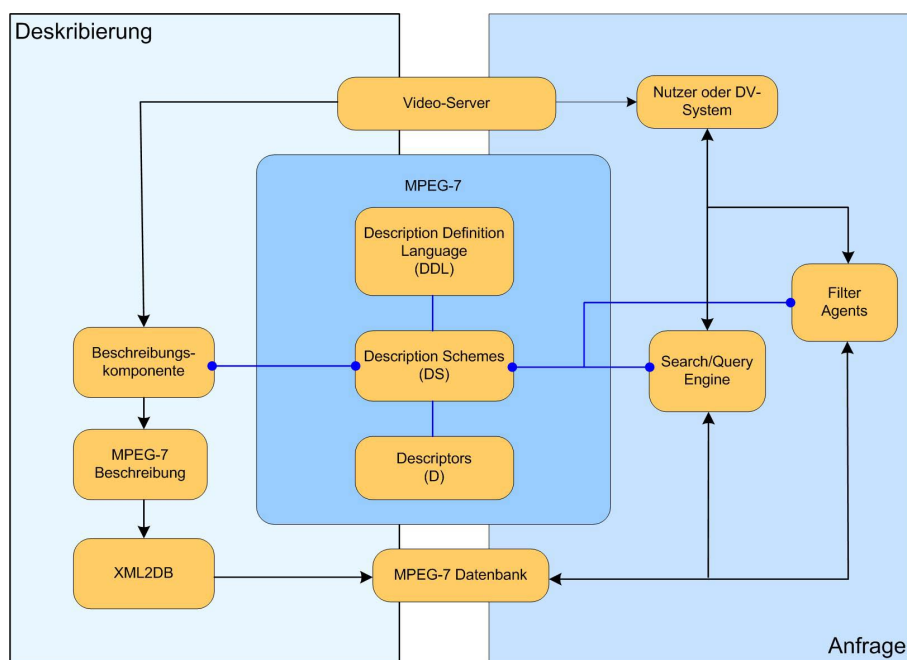
**Inhalt:** Handlungsorte und –personen, Genre, visuelle oder textliche Zusammenfassungen des Inhaltes (Keyframes, Abstracts)

**Produktion:** Titel, Autor, Mitwirkende

**Medieninformationen:** Speicherort des Videos, Verschiedene Varianten des Videos, Kodierungsformat, Bitrate

**Nutzung:** Nutzungsbeschränkungen, Zielgruppen, Related Material

Diese Eigenschaften können innerhalb MPEG-7 mit Deskriptoren und Description Schemes aus den Tools des MDS (siehe Abschnitt 4.1) beschrieben werden. Die im Anwendungskontext notwendigen Deskriptoren und Description Schemes dienen dann sowohl zur Erzeugung von Beschreibungen der audiovisuellen Daten als auch zur Unterstützung der Suche auf ihnen. Das in Abbildung 5-1 angegebenen Schema illustriert diese Einbindung von MPEG-7 in das Content Management audiovisueller Daten.



**Abbildung 5-1:** Content-Management mit MPEG-7

Ausgangspunkt für die Beschreibung sind auf einem Videoserver in digitaler Form vorliegende Videos. Diese werden mittels einer Beschreibungskomponente unter Beachtung



der zugrundeliegenden Deskriptorenmenge beschrieben. Die so entstandenen MPEG-7 Beschreibungen werden zur Verwaltung und Zugriffsunterstützung in einer Datenbank gespeichert. Dies realisiert die XML2DB-Komponente.

Über eine Search/Query Engine können durch den Nutzer des Systems Suchanfragen gestellt werden. Diese werden unter Beachtung der verwendeten MPEG-7 Elemente in Datenbankabfragen an die MPEG-7 Datenbank umgeformt. Die durch die Datenbank gelieferte Ergebnismenge wird ggf. gerankt und dem Nutzer als Resultatsmenge in einer ihm verständlichen Form präsentiert. Ein Bestandteil eines einzelnen Resultates sollte ein Verweis auf das entsprechende Videosegment sein, über den der Nutzer dann direkt auf den gewünschten Inhalt zugreifen kann.

Um die Effektivität der Suche auch schon bei einfachen Anfragen zu erhöhen, oder sogar automatische Anfragen zu ermöglichen, können mittels MPEG-7 neben den Videodaten auch Nutzerpräferenzen beschrieben werden. Hierfür stehen durch die UserInteraction Tools umfangreiche Beschreibungsmöglichkeiten zur Verfügung (siehe Abschnitt 4.2). Auch hier ist für konkrete Anwendungen eine Auswahl relevanter Beschreibungselemente zu wählen. Durch die Beschreibung der bevorzugten inhaltlichen Eigenschaften durch MPEG-7 Deskriptoren, welche mit der Beschreibung der Inhalte der Videodaten korrespondiert, ist eine Filterung relevanter Inhalte aus einer großen Menge von audiovisuellen Daten möglich. Diese kann auch automatisch ohne zusätzliche Nutzereingaben durch eine Filteragenten ausgeführt werden.

Für die Umsetzung solch einer inhaltsbasierten personalisierten Suche auf audiovisuellen Dokumenten ist eine geeignete Deskriptormenge sowohl für die Beschreibung der Daten als auch für die Beschreibung der Nutzerinteressen zu bestimmen. Dazu wird das MPEG-7 Schema auf die notwendigen Deskriptoren reduziert. Im folgenden Abschnitt werden zunächst Möglichkeiten zur Reduzierung des MPEG-7 Schemas aufgezeigt. In den sich daran anschließenden Abschnitten werden die konkreten Schemas für die Beschreibung der Daten und der Nutzerinteressen sowie die in dieser Arbeit genutzte Art der Verwaltung der MPEG-7 Daten vorgestellt.

### **5.1.1 Reduzierung des Schemas**

MPEG-7 stellt eine sehr große Anzahl von Deskriptoren und Description Schemes zur Beschreibung audiovisueller Daten zur Verfügung. Für konkrete Anwendungen ist ein

Bruchteil dieser Deskriptoren zur adäquaten Beschreibung ausreichend. Es empfiehlt sich also die Anwendung eines reduzierten Schemas. Durch ein reduziertes Schema wird unter anderem die Abbildung der Dokumentenstrukturen auf für die Verwaltung notwendige Zugriffsstrukturen, wie Datenbankschemata, erleichtert. Außerdem können speziell für das reduzierte Schema entwickelte Deskribierungswerkzeuge eingesetzt werden, durch die der Beschreibungsprozess übersichtlicher gestaltet und damit erleichtert werden kann.

Das durch Reduzierung gewonnene Teilschema muss ein Unterschema des Ausgangsschemas darstellen d.h. alle auf Basis des neuen Schemas generierbaren Dokumente (Instanzen des Schemas) müssen auch dem Ausgangsschema genügen. Die Menge der beschreibbaren Instanzen eines vom Schema S reduzierten Schemas S' ist eine echte Teilmenge der Menge der beschreibbaren Instanzen von S. Die mit S# erzeugbaren Dokumente sind dann MPEG-7 konform. Speziellere MPEG-7 Beschreibungen können jedoch nicht ohne Informationsverlust in das auf S' aufbauende System integriert werden.

Ein reduziertes Teilschema kann durch das Weglassen optionaler Elemente, also einer Reduzierung des Beschreibungsumfanges, die Verschärfung von Bedingungen und die Beschränkung von Kardinalitäten (Heraufsetzen der unteren Grenze, Herabsetzen der oberen Grenze der Anzahl des möglichen Auftretens eines Elementes<sup>1</sup>) erhalten werden. Da die MPEG-7 DDL auf XML-Schema basiert können zur Reduzierung des MPEG-7 Schemas Reduktionsansätze von XML-Schema genutzt werden. In Tabelle 5-1 sind Reduktionsmöglichkeiten für die Typdeklaration in XML-Schema nach [XMLS01] angegeben.

Basis	Einschränkungen	Bemerkung
	default="1"	setzen eines Defaultwertes, wo keiner war
	fixed="100"	Setzen eines festen Wertes, wo keiner war
	type="string"	Setzen eines Typs, wo keiner war
die folgenden Zahlengruppen beziehen sich auf Grenzen (minoccurs,maxoccurs)		
(0,1)	(0,0)	Weglassen einer optionalen Komponente
(0,unbounded)	(0,0) (0,24)	weitere Beschränkungen von Kardinalitäten
(1,9)	(1,6) (2,9) (3,3)	
(1,unbounded)	(1,9) (2,3) (4,unbounded)	

**Tabelle 5-1:** Restriktionsmöglichkeiten in XML-Schema (nach [XMLS01])

<sup>1</sup> Das Weglassen eines optionalen Elementes ist ein Spezialfall hiervon. Die obere Grenze des möglichen Auftretens wird auf 0 herabgesetzt.

Mit Hilfe dieser Reduktionen wird das MPEG-7 Schema im folgenden den Anforderungen des Anwendungsszenarios entsprechend angepasst. Dabei wird sowohl für die Beschreibung der Videos als auch für die Beschreibung der Nutzerinteressen ein reduziertes Schema mit den notwendigen Deskriptoren entwickelt.

### 5.1.2 Deskriptormenge Videodaten

Um Anfragen auf Videodaten stellen zu können, müssen die Eigenschaften von Videos anhand von Metadaten beschrieben werden. Im vorliegenden Anwendungsszenario soll eine inhaltsbasiert personalisierte Suche auf Nachrichtenbeiträgen mit verschiedenen Endgeräten unterstützt werden. Als Suchkriterien sollen Sender, Sprecher, Nachrichtenkategorie, und Sendezeit sowie eine Texteingabe zur Suche auf Schlagworten und Titel des Beitrages unterstützt werden. Damit müssen folgende Eigenschaften der Videos mittels geeigneter Deskriptoren und Description Schemes beschrieben werden:

- unterschiedliche Formate des Videos
- zeitliche Segmentierung einer Nachrichtensendung in einzelne Beiträge
- Sprecher , Sender, Sendezeit und Titel eines Beitrages
- Art eines Beitrages (Genre)
- Textannotationen zu einem Beitrag mit Inhaltsangabe und Schlagworten

Zur Visualisierung der einzelnen MPEG-7 Schemasegmente wird eine in [Rust01] vorgeschlagene Darstellung in Form von Baumgrafiken genutzt, die nicht die gesamte Beschreibungsfunktionalität von XML-Schema ausdrücken kann, jedoch einen Kompromiss zwischen Ausrucksfähigkeit und Übersichtlichkeit bildet.

In den einzelnen Knoten befinden sich dabei jeweils nur die in schemakonformen Dokumenten erzeugbaren Attribut- und Elementnamen. Optionalität von Elementen wird mittels gestrichelter Linien dargestellt, Mehrwertigkeit durch ein besonderes Linienende. Elementstrukturierungen mittels *sequence* oder *choice* werden normalisiert auf einer Ebene dargestellt. Typinformationen werden vernachlässigt. Die Blätter eines Segmentbaumes bestehen im allgemeinen aus Elementen simplen Typs. Komplexe Blattelemente sind durch einen gestrichelten Rand und eine andere Färbung gekennzeichnet.

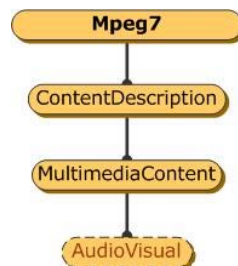
## Grundlegende Beschreibungselemente

Ausgangspunkt jeder MPEG-7 Beschreibung ist das Root-element *Mpeg7*. Um Inhaltsbeschreibungen der Multimediaeinheit Video anzulegen, ist als Top-level Element *ContentDescription* des Types *ContentEntity* zu wählen. Dieses besitzt das Element *MultimediaContent*. Mit diesem DS kann eine Instanz multimedialen Inhaltes beschrieben werden. Da für diese Arbeit Videos beschrieben werden sollen, muss *MultimediaContent* vom Typ *AudioVisualType* sein. Damit besitzt es das Element *AudioVisual*. Mit Hilfe des *AudioVisual* DS können nun die Eigenschaften des Videos beschrieben werden. Eine MPEG-7 Beschreibung eines Videos ist also wie folgt aufgebaut:

```
<Mpeg7 xmlns=http://www.mpeg7.org/2001/MPEG-7_Schema xml:lang="en"
type="complete">
  <ContentDescription xsi:type="ContentEntityType">
    <MultimediaContent xsi:type="AudioVisualType">
      <AudioVisual>
        ...
      </AudioVisual>
    </MultimediaContent>
  </ContentDescription>
</Mpeg7>
```

**Beispiel 5-1:** Grundschemata einer Video-Beschreibung mit MPEG-7

Die Abbildung 5-2 stellt diese Struktur des Grundschemas einer MPEG-7 Beschreibung zur besseren Anschaulichkeit graphisch dar.



**Abbildung 5-2.** Reduziertes Teilschema MPEG-7

## Zeitliche Segmentierung

Für die Beschreibung einzelner Nachrichtenbeiträge ist eine zeitliche Segmentierung der Videos notwendig. Die Abbildung von zeitlicher Segmentierung audiovisueller Daten erfolgt mit Hilfe des *TemporalDecomposition* DS welches innerhalb eines *AudioVisual* DS

instanziiert werden kann. Die so entstandenen Segmente werden mittels des `AudioVisualSegment` DS beschrieben. Dabei wird auch das gesamte Video als ein `AudioVisualSegment` angesehen. Eine *TemporalDecomposition* kann auch innerhalb eines `AudioVisualSegment` instanziiert werden, und aus beliebig vielen audiovisuellen Segmenten bestehen. So kann rekursiv eine beliebig tiefe Schachtelung vorgenommen und eine beliebig tiefe Segmenthierarchie ausgedrückt werden. Im vorliegenden Anwendungsszenario wird die Schachtelung nur einmal vorgenommen. Das audiovisuelle Segment „gesamte Nachrichtensendung“ wird in die audiovisuellen Segmente „Beitrag“ unterteilt. Sowohl die Eigenschaften der Sendung als auch die des Beitrages können nun über das Element `AudioVisualSegment` beschrieben werden. Eine zeitliche Unterteilung eines Videos in zwei Beschreibungsebenen ist im Beispiel 5-2 dargestellt.

```
<AudioVisual>
  <TemporalDecomposition>
    <AudioVisualSegment>
      <TemporalDecomposition>
        <AudioVisualSegment/>
        <AudioVisualSegment/>
        <AudioVisualSegment/>
      </TemporalDecomposition>
    </AudioVisualSegment>
  </TemporalDecomposition>
</AudioVisual>
```

#### Beispiel 5-2: Temporale Segmentierung

Für die zeitliche Strukturierung audiovisueller Daten ist eine Angabe der Zeitdauer (Länge) und des Zeitpunktes des Beginns eines zeitlichen Segmentes notwendig. Für Zeitangaben steht in MPEG-7 der Datentyp *MediaTimeType* zur Verfügung, der als Komponenten eine Zeitpunkt und optional eine Zeitspanne enthält. Zeitpunkte können dabei auf drei unterschiedliche Arten angegeben werden:

**MediaTimePoint:** fester Zeitpunkt als Datum nach dem gregorianischen Kalender

**MediaRelTimePoint:** Zeitpunkt definiert als zeitlicher Offset ausgehend von einem vorher definierten Zeitpunkt (timeBase)

**MediaRelIncrTimePoint:** Zeitpunkt ausgehend von einer time base unter Nutzung eines vordefinierten Intervalls (time unit)

Zeitspannen können mittels *MediaDuration* durch die Angabe des Zeitintervalls in Tagen  $nD$  ( $n$  days) sowie einer Unterteilung des Tages ( $TnHnMnS$ ) welche auch eine Angabe zur Unterteilung der Sekunden ( $nS$ ) in eine Anzahl ( $nN$ ) von Teilen (fractions) von Sekunden ( $nF$ ) beinhaltet, zusammengefasst wird ein Zeitintervall also in der Form  $nDTnHnMnSnNnF$  angegeben.

Die Unterteilung der Videos in Zeitintervalle wurde in der vorliegenden Arbeit mit dem Ricoh MovTool<sup>2</sup> vorgenommen. Dabei wurden zur Bestimmung des Startzeitpunktes der Typ *MediaRelTimePoint* und zur Angabe der Dauer eines Segments *MediaIncrDuration* genutzt. Die Anzahl von fractions einer Sekunde ist dabei 30, da MPEG-1 Videos, die eine Framerate von 30 Frames pro Sekunde aufweisen, beschrieben werden. Der Startzeitpunkt jedes Segmentes ist relativ zum Beginn des gesamten Videos angegeben. Dadurch kann auf die einzelnen Segmente framegenau zugegriffen werden.

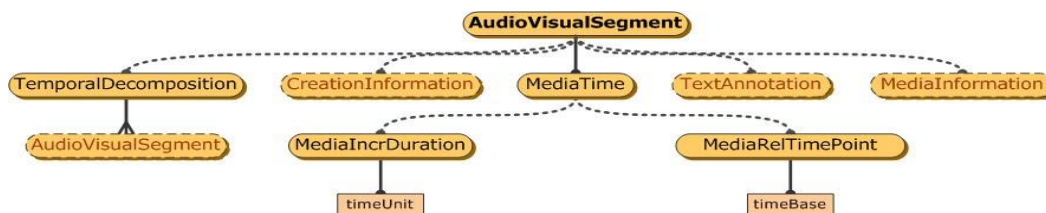


Abbildung 5-3: Reduziertes Teilschema AudioVisualSegment DS

Die Abbildung 5-3 zeigt die Struktur eines AudioVisualSegments mit zeitlicher Segmentierung und Angabe der Medienzeit sowie weiteren im folgenden beschriebenen Elementen.

### Lokalisierung des Videos/ Angabe von Versionen

Für die Verknüpfung der beschreibenden Daten mit dem eigentlichen Videos muss ein Verweis auf das Video in der Beschreibung vorhanden sein. Dieser kann innerhalb des Media Information DS angegeben werden. Ein Video sollte in mehreren Versionen verfügbar sein, so dass mit unterschiedlich leistungsstarken Geräten/Netzwerkverbindungen zugegriffen werden kann. Zu Beschreibung der Eigenschaften verschiedener Versionen des Videos kann das *MediaProfile* DS innerhalb von *MediaInformation* mehrfach instanziiert werden. Jede einzelne Version wird dann in einem Mediaprofile beschrieben. Innerhalb von MediaProfile werden folgende Eigenschaften beschrieben:

**Dateiformat** der Version: über das DS Fileformat innerhalb des MediaFormat DS. anzugeben ist über das Attribut *href* ein Term aus einem ClassificationScheme (siehe Abschnitt 4.2.1) im Element *Name* wird der Name des Terms angegeben

<sup>2</sup> Siehe <http://src.ricoh.co.jp>

**Bitrate** über das Element *MediaFormat/Bitrate* angegeben in bit/s

**Kodierungsinformationen** innerhalb des *VisualCoding* DS können Informationen zur Kodierungsart des Videos angegeben werden: die Framegröße über die Attribute *height* und *width* des Elements *Frame* und der Name des Kodierungsformat über das Element *Name* im DS *Format*. Zur eindeutigen Bezeichnung/Bedeutungsbestimmung wird im Attribut *href* ein *ClassificationScheme* angegeben aus dem der Name entnommen wurde.

**Speicherungsart** der Version: im Element *MediaLocator/MediaUri* innerhalb des *MediaInstance* DS. In *MediaInstance* kann für die Version außerdem mit dem Element *InstanceIdentifier* eine dokumentweit eindeutige ID festgelegt werden.

Die Struktur des *MediaInformation* Ds zeigt Abbildung 5-4 .

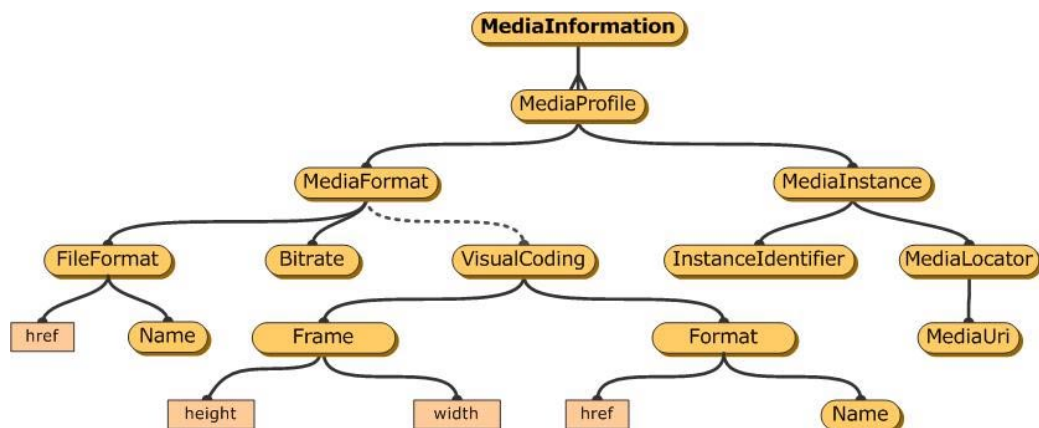


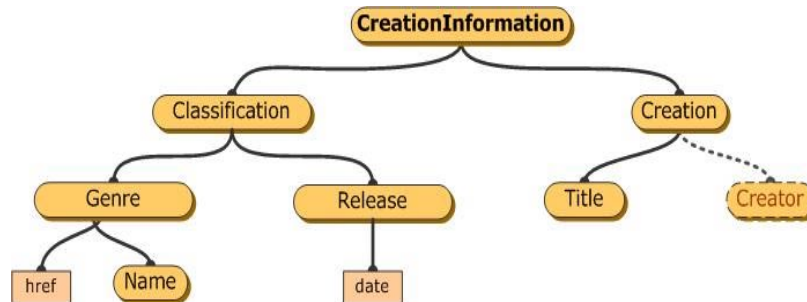
Abbildung 5-4 reduziertes Teilschema *MediaInformation* DS

Das Element *VisualCoding* ist als optional beibehalten worden. An dieser Stelle wurde das Schema nicht verändert, da auch die Möglichkeit der Angabe nicht-visueller Versionen (Text) für die Übertragung auf leistungsschwache Geräte gegeben sein soll.

### Angabe von Produktionsinformationen

Relevante Angaben, die Produktion des Videos betreffen, sind im vorliegenden Anwendungsszenario die Art der Information (Genre), der Name des Sprechers der Nachrichtensendung/des Nachrichtenbeitrages, sowie der Titel des Beitrages/der Sendung.

Diese Eigenschaften können innerhalb des CreationInformation DS beschrieben werden. Dazu stehen dort die Elemente Classification DS, zur Zuordnung zu einer Kategorie, und Creation DS, mit dessen Hilfe Eigenschaften, die die Herstellung betreffen, beschrieben werden können, zur Verfügung. Die Struktur des CreationInformation DS ist in Abbildung 5-5 dargestellt.



**Abbildung 5-5:** reduziertes Teilschema CreationInformation DS

Die für die Anwendung relevanten Informationen sind dabei wie folgt beschrieben:

**Titel** innerhalb des Creation DS mit Hilfe des Elementes *Title*

**Sprecher** über das Creator DS ebenfalls innerhalb des Creation DS. Ein Creator kann dabei vom Typ *Person* oder *Organization* sein. Für eine Person wird dabei im hier genutzten Anwendungsszenario ein Namen und ein Vorname sowie optional ein Titel angegeben, für eine Organisation ein Name. Über das Element *Role* kann dem Creator die Rolle, die er im Produktionsprozess spielt zugeordnet werden. Über die Angabe eines ClassificationSchemes mittels des Attributes *href* wird die Eindeutigkeit der Bezeichnung gewährleistet. Die Struktur des Creator DS veranschaulicht Abbildung 5-6.

**Art des Beitrages:** durch das Element *Genre* innerhalb des Classification DS. Der Name des Genres steht im Element *Name* und stammt aus dem im Attribut *href* angegebenen ClassificationScheme (CS). Hierbei bietet sich ein hierarchisch aufgebautes CS an. Im MPEG-7 Standard wird als Beispiel eines für Genre einsetzbaren CS das von TV-Anytime [TVA] entwickelte, Content CS genannte, Classification Scheme angegeben, welches auch in dieser Arbeit verwendet wird.



**Sendezeitpunkt:** durch das Attribut *date* des Elements *Release* innerhalb des Classification DS.

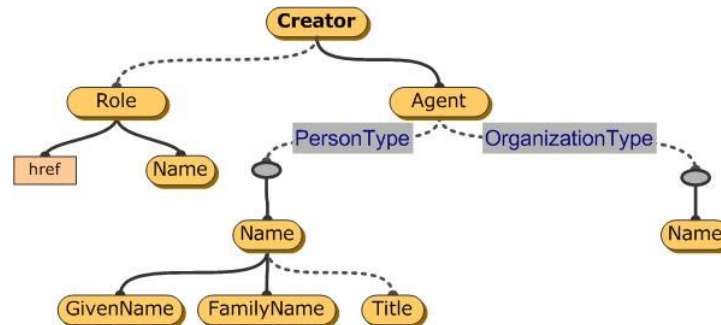


Abbildung 5-6: reduziertes Teilschema Creator DS

### Angabe von Nutzungsinformationen

Angaben zur Nutzung der audiovisuellen Daten können im UsageInformation DS gemacht werden. Für die hier vorgestellte Anwendung ist dabei der **Name des Senders**, der die Nachrichtensendung ausstrahlt, interessant. Dieser kann mit dem Element *Distributor* aus dem Availability DS welches innerhalb des UsageInformation DS instanziiert werden kann, angegeben werden. Der Distributor ist dabei vom abstrakten Typ *Agent*, kann also wie ein Creator sowohl vom Typ *Person* als auch vom Typ *Organization* sein.

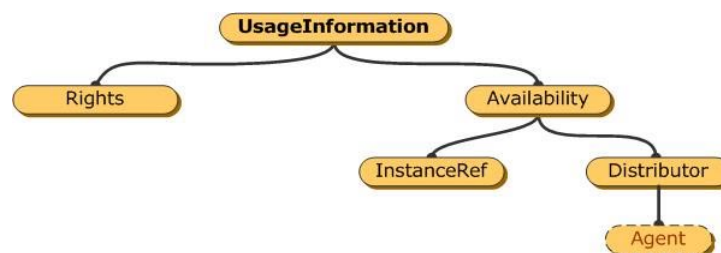


Abbildung 5-7: Reduziertes Teilschema UsageInformation DS

### Angabe zusätzlicher Inhaltsinformationen

In dieser Arbeit soll auch eine textbasierte Suche auf Schlagworten und Zusammenfassungen unterstützt werden. Diese zusätzliche Inhaltsbeschreibungen sind mit dem TextAnnotation DS innerhalb eines AudioVisualSegment wie folgt angebar.

**Schlagnworte:** mit dem KeywordAnnotation DS, welches mehrere Schlagwörter als *Keyword*-Elemente enthalten kann

**Inhaltszusammenfassungen:** über das Element *FreeTextAnnotation*

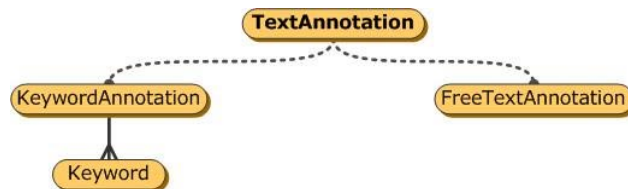


Abbildung 5-8: Reduziertes Teilschema TextAnnotation DS

### 5.1.3 Deskriptormenge Nutzerpräferenzen

Durch das UserPreferences DS ist es möglich Nutzerpräferenzen innerhalb des MPEG-7 Standards zu beschreiben, die sowohl den Inhalt von Daten als auch deren Darstellung betreffen (vgl. Abschnitt 4.2.1). Die Übereinstimmung zwischen der Beschreibungen der Nutzerpräferenzen und der Inhalte ermöglicht eine genaue und effiziente personalisierte Datennutzung. Im Rahmen dieser Arbeit werden nur die auf den Inhalt bezogenen Präferenzen berücksichtigt. Durch die Angabe inhaltsbezogener Präferenzen soll eine contentbasiertes Filterung von Informationen (vgl. Abschnitt 2.3) ermöglicht werden. Die Eigenschaften der Nachrichtenbeiträge, zu denen der Nutzer Präferenzen angeben kann, nach denen ihm personalisierte Ergebnisse geliefert werden, sind hierbei: Nachrichtenkategorie (Genre) sowie Sprecher und Sender des Nachrichtenbeitrages.

Diese Präferenzen können innerhalb des UsagePreference DS mittels des FilteringAndSearchPreferences DS angegeben werden. Dazu stehen dort die Komponenten *CreationPreferences*, *ClassificationPreferences* und *SourcePreferences* zur Verfügung. Die einzelnen Präferenzen können wie folgt angegeben werden

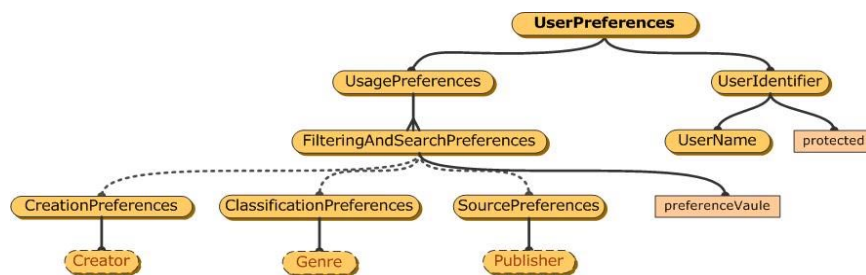
**Nachrichtenkategorie:** innerhalb der ClassificationPreferences über das Element *Genre*, welches dem im Classification DS (siehe 5.1.2) entspricht

**Nachrichtensprecher:** innerhalb der CrationPreferences über das Element *Creator*, welches dem im Creation DS (siehe 5.1.2) entspricht.

**Nachrichtensender:** über das Element *Publisher* innerhalb der *SourcePreferences*. *Publisher* ist dabei vom abstrakten Typ *Agent* und kann so eine *Organization* oder eine *Person* sein. Hier stimmt die Bezeichnung nicht mit dem im Usage DS instanziierten Element *Distributor* überein. Im MPEG-7 Standard ist bei der Semantikdefinition der *SourcePreferences* ([MPEG01a] S.562) jedoch der *Distributor* des audiovisuellen Inhaltes als mögliches Beschreibungsobjekt von *Publisher* angegeben. Beim Vergleich von Dokumenten und Präferenzen ist auf diese unterschiedliche Bezeichnung zu achten.

Der Aufbau des Description Schemes für Nutzerpräferenzen ist in Abbildung 5-9 dargestellt.

Der Name des Nutzers, für den die Präferenzen beschrieben werden, wird innerhalb des Elementes *UserIdentifier* mittels *UserName* angegeben. Das Attribut *protected* gibt an, ob das Nutzerprofil Dritten zugänglich gemacht werden kann.



**Abbildung 5-9:** Reduziertes Schema UserPreferences DS

Zur Angabe mehrerer Präferenzen eines Nutzers können die FilteringAndSearch Preferences mehrfach instanziiert werden. Für die einzelnen Präferenzen können mit dem Attribut *preferenceValue* Wertigkeiten angegeben werden. Der *preferenceValue* kann dabei Integerwerte zwischen  $-100$  und  $100$  annehmen, welche angeben, wie wichtig die einzelne Präferenz für den Nutzer ist. Positive Werte bedeuten eine Bevorzugung von Objekten mit den angegebenen Eigenschaften, ist der Wert negativ, so sind Objekte mit der entsprechenden Eigenschaft mehr oder weniger nicht gewünscht. Die Maximalwerte sollen in dieser Arbeit signalisieren, dass der Nutzer bei der Suche nur Dokumente die den angegebenen Eigenschaftswert haben bzw. nicht haben (*preferenceValue*  $100$  bzw.  $-100$ ) erhalten will. Hier wandeln sich die Präferenzen in Bedingungen (Conditions, siehe Abschnitt 2.1), die schon bei der Anfrage berücksichtigt werden, was einer Filterung mittels

booleschem Retrieval entspricht. Ansonsten fließen die Wertigkeiten der Präferenzen in das Ranking der Suchergebnisse ein.

#### **5.1.4 Verwaltung der MPEG-7 Beschreibungen**

Um effizient auf die MPEG-7 Beschreibungen der Videos und Nutzerpräferenzen zugreifen zu können, empfiehlt sich die Speicherung in einer Datenbank. Die Beschreibungen liegen als XML-Dokumente vor, so dass Techniken der Speicherung von XML-Dokumenten in Datenbanken angewendet werden können.

Für die Abbildung von XML Dokumentenstrukturen auf Datenbanken existiert eine Reihe unterschiedlicher Ansätze. Dabei kann die Art der verwendeten Datenbank, die Art der Anfrage auf die Daten sowie die Art der Abbildung der XML-Datenstrukturen auf die konkreten Datenbankstrukturen variieren.

Bourret unterscheidet in [Bour02] drei Datenbanktypen zur Speicherung von XML-Daten: traditionelle Datenbanken, XML-enabled Datenbanken und native XML-Datenbanken. Bei dem Einsatz von traditionellen Datenbanken wie relationalen, objektorientierten und hierarchischen Datenbanken für die Verwaltung von XML-Dokumenten müssen spezielle Parser zum Einfügen der Daten aus XML-Dokumenten in die Datenbank genutzt werden. Sind diese bereits in die Funktionalität eines traditionellen DBMS integriert, so spricht man von XML-enabled Datenbanken. Native XML-Datenbanken sind speziell für die Speicherung von XML-Dokumenten entwickelt. In dieser Arbeit wird eine traditionelle, relationale Datenbank zur Verwaltung der MPEG-7 Daten genutzt. Dabei müssen die XML-Dokumentstrukturen auf relationale Datenbankstrukturen abgebildet werden. Auch hier können unterschiedliche Verfahren zur Anwendung kommen.

So werden in [STH+99] detailliert verschiedene Möglichkeiten der Abbildung von XML auf relationale Strukturen auf Grundlage von DTDs beschrieben. Eine mögliche Abbildung von mittels XML-Schema beschriebenen XML-Dokumenten wird in [VaVa01] vorgestellt. Dabei wird jedoch für jeden komplexen Elementtyp aus dem Schema eine eigene Tabelle in der Datenbank angelegt, was zu einer sehr großen Anzahl von Tabellen führt. So müssen nach diesem Ansatz schon bei einer Anzahl von 5 komplexen Elementtypen insgesamt 11 Tabellen angelegt werden um das XML-Schema abzubilden. In dieser Arbeit wird ein von Rust [Rust01] genutzter Ansatz zur Abbildung von XML-Dokumentenstrukturen auf relationale Datenbanken eingesetzt. Dieser basiert auf einer Abbildung von DTDs auf

relationale Datenbanken der in [KlMe00] vorgeschlagen wird, und bezieht die Erweiterungen von XML-Schema ein.

### **Prinzip der Abbildung von MPEG-7 Beschreibungen auf relationale Datenbanken**

Bei der hier genutzten Abbildung von XML-Dokumentstrukturen auf relationale Datenbankstrukturen wird vom Wurzelement des XML-Dokumentes ausgegangen. Diesem wird ein Relationenschema zugeordnet. Bei der Abbildung von Attributen und Kindelementen sind Instanzierungskardinalitäten, also die mögliche Anzahl des Auftretens eines Elementes als Kindknoten im XML-Baum, zu beachten.

Darf ein Element nur höchstens ein mal als Kindknoten vorkommen (Kardinalitäten [0;1] oder [1;1]), so kann dieser Knoten rekursiv auf ein Attribut des Relationenschema des Elternknotens abgebildet werden. Ist das Auftreten des Kindelementes optional ( [0;1]), so sind im entsprechenden Datenbankattribut Nullwerte zugelassen.

Kann ein Element mehrfach instanziiert werden ([0,n] oder [1;n]), so muss für dieses Element ein eigenes Relationenschema erstellt werden. Integrität der Daten wird dabei über eine Fremdschlüsselbeziehung erreicht. Dazu muss das Elternelement eine eindeutige dokumentenübergreifende Identität besitzen, die in der Kindrelation als Fremdschlüssel auftaucht. Im MPEG-7 Standard sind nur dokumentweit eindeutige Bezeichner vorgesehen, so dass eine dokumentübergreifende Objektidentität beim Einfügen in die Datenbank durch das System erzeugt werden muss. Der sich in der Entwicklung befindende MPEG-21 Standard soll dieses Problem im Teil 3 „Digital Item Identification and Description“ [MPEG01b] durch eine eindeutige Identifikation der Digital Items (in unserem Fall sowohl ganzer Videos als auch Teile von Videos) lösen.

#### Besonderheiten XML-Schema

Zusätzlich zu diesem, für Strukturbeschreibungen mit DTDs entwickeltem, Mechanismus müssen bei Abbildungen vom XML-Schema Beschreibungen einige der dort eingesetzten Erweiterungen beachtet werden. Während Strukturen in DTDs anhand von Elementnamen ausgedrückt werden und nur wenige elementare Datentypen zur Verfügung stehen, erlaubt XML-Schema die Definition von eigenen Typen. Es können außerdem

Vererbungsmechanismen für den Aufbau von Typhierarchien eingesetzt werden. Dies ist auch im vorliegenden reduzierten Schema der Fall.

Hier sind *PersonType* und *OrganizationType* vom abstrakten Datentyp *AgentType* abgeleitet. Dieser wird innerhalb des *Creator* DS und innerhalb von *Distributor* für das Element *Agent* genutzt. Für die Abbildung dieser beiden Typen bestehen zwei Möglichkeiten:

**Abbildung auf einzelne Schemas:** *OrganizationType* und *PersonType* werden auf einzelne Relationen abgebildet. Die Integrität wird wieder über Fremdschlüsselbeziehung zur Elternrelation gesichert. Hat der abstrakte Typ viele vererbte Attribute, so könnten auch drei einzelne Relationenschemas gebildet werden. Dies ist aber bei *AgentType* nicht der Fall. Er besitzt keine vererbten Attribute.

**Bildung eines gemeinsamen Schemas:** Beide Typen werden auf ein gemeinsames Schema abgebildet wobei die jeweils nicht benutzten Attribute mit Null-Werten belegt werden.

Die Bildung eines gemeinsamen Schemas kann insbesondere dann vorteilhaft sein, wenn Elemente mit abstraktem Typ als höchsten einmal instanziierebare Kindelemente anderer Elemente vorkommen. Die Attribute können dann in das Elternschema übernommen werden.

Die im XML-Schema selbst definierten Elementtypen können innerhalb des Schemas für Elemente mit unterschiedlichen Namen verwendet werden. So wird der (abstrakte) Typ *AgentType* sowohl für das Element *Creator* innerhalb des *Creation* DS als auch für das Element *Distributor* innerhalb von *Availability* DS genutzt (siehe Abb. 5-5 und 5-7). Für die Abbildung von Elementtypen auf Datenbankschemas stehen ebenfalls zwei Möglichkeiten zur Auswahl:

**Abbildung anhand Elementnamen:** Hierbei werden die Typen der Elemente nicht bei der Abbildung auf Relationenschemas berücksichtigt. Für jedes Element wird ein eigenes Relationenschema angelegt.

**Abbildung anhand Elementtypen:** Hier werden Informationen zum Typ bei der Abbildung auf Relationenschemas genutzt. Es wird insgesamt nur ein Relationenschema für beide Elemente erstellt. In unserem Beispiel wäre dies das Relationenschema *Agent*. Probleme entstehen, wenn zwei Elemente gleichen Typs gemeinsam als Kindelemente vorkommen. Hier müssen für beide Elemente neue

Relationenschemas eingefügt werden, da sonst die Attribute des gemeinsamen Typs den Elementen nicht eindeutig zugeordnet werden können.

### Konkretes Datenbankschema

Bei der Bildung des Datenbankschemas aus dem reduzierten MPEG-7 Schema wurde von oben vorgestellten Grundprinzip der Abbildung ausgegangen. Im MPEG-7 Schema für die Videodaten sind die Elemente Creator, AudioVisualSegment und MediaInformation und Keyword mehrfach instanzierbar. Für diese müssen somit eigene Tabellen angelegt werden. Die in der Hierarchie unter AudioVisualSegment stehenden Elemente, die nicht mehrfach instanziiert werden können, werden als Attribute in die Tabelle AudioVisualSegment aufgenommen. Die möglichen konkreten Elementtypen des abstrakten Typs Agent, Person und Organization wurden in einem Schema zusammengefasst. Der Typ Agent wird außerdem in den Typen Distributor und Creator genutzt (in letzterem erweitert um das Element Role). Hier wurde die Abbildung anhand Elementnamen gewählt. Somit ergibt sich das in Abbildung 5-10 gezeigte Datenbankschema für die Videodaten.

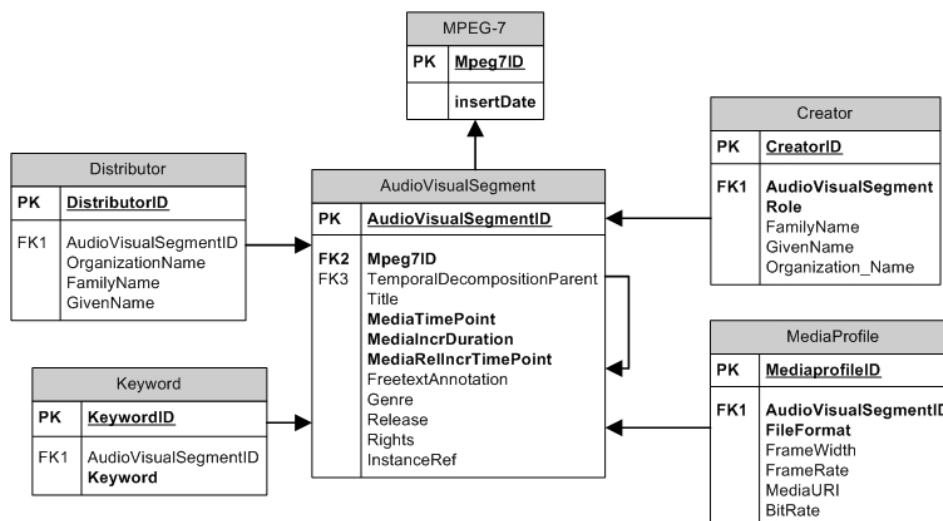
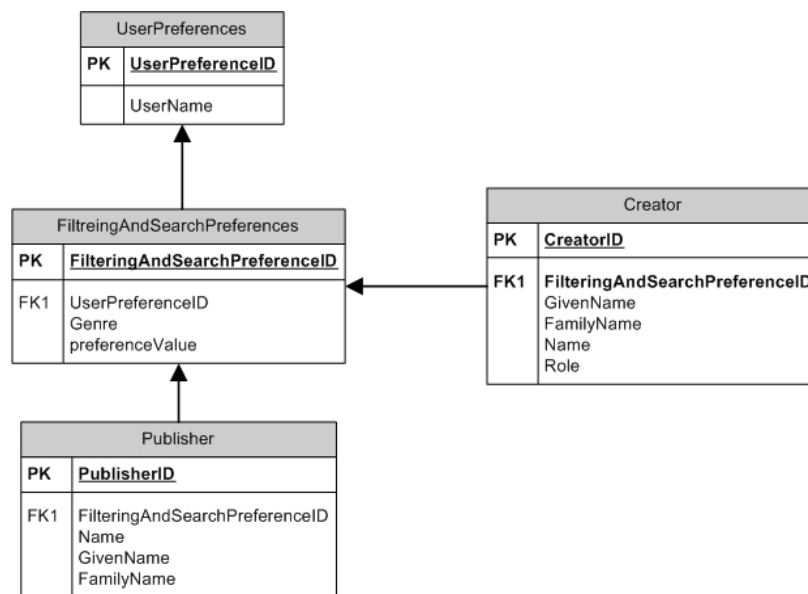


Abbildung 5-10: Datenbankschema für Videodaten

Bei der Erstellung des Datenbankschemas für die Nutzerpräferenzen wurde auf gleiche Weise vorgegangen, wie beim Datenbankentwurf für die Videodaten. Hier können die FilteringAndSearchPreferences innerhalb der UsagePreferences mehrfach instanziiert

werden (vgl. Abbildung 5-8), und werden somit in ein eigenes Relationenschema abgebildet. Bei der Abbildung des AgentTyps der Creator und Publisher zugeordnet ist, wurde auf gleiche Weise wie bei Erstellungen des Schemas für die Videodaten vorgegangen, so dass die Elemente Creator und Publisher auf eigene Relationen mit den Attributen aus Person und Organization abgebildet wurden. Das Ergebnis der Abbildung des Schemas für die Nutzerpräferenzen zeigt Abbildung 5-11.



**Abbildung 5-11:** Datenbankschema für Nutzerpräferenzen

Wie schon an anderer Stelle erwähnt sind auch andere als die hier vorgenommene Abbildung der reduzierten MPEG-7 Schemas auf relationale Datenbankschemas möglich. Es lassen sich also neben den vorliegende auch andere Datenbankschemas generieren, die die gleichen MPEG-7 Beschreibungen abbilden. Dies ist im weiteren Verlauf der Arbeit in Betracht zu ziehen.

## 5.2 Einbindung in XML-Framework

Im vorhergehenden Abschnitt wurden mit der Definition von Teilschemas und der Abbildung dieser auf eine relationale Datenbank Grundlagen für ein personalisiertes Content Management auf Basis von MPEG-7 für das gegebene Beispielszenario, die inhaltsbasierte Suche auf Nachrichtensendungen, geschaffen. Ziel dieser Arbeit ist die Einbindung von auf



MPEG-7 basierenden Anwendungen in eine erweiterbare Framework-Architektur um eine universelle Multimedia Contentverwaltung zu ermöglichen. Eine solche offene erweiterbare Architektur wurde im Rahmen des XPEA<sup>1</sup>-Projektes im ZGDV e.V. Rostock entwickelt und soll hier als Grundlage dienen. Sie wird im folgenden überblicksartig vorgestellt bevor die notwendigen Erweiterung zur Durchführung einer personalisierten Informationsfilterung auf audiovisuellen Daten dargestellt werden.

### 5.2.1 XPEA-Architektur

Mit der auf einem in [Ditt01] entwickelten Prototyp basierenden XPEA-Architektur steht ein flexibles erweiterbares Framework auf XML-Basis für universellen Zugriff auf multimediale Dokumentenstrukturen zur Verfügung. Es bietet die Möglichkeit der Unterstützung einer Reihe von Anwendungen aus unterschiedlichsten Bereichen auf verschiedenen Endgeräten, durch eine serverseitige anwendungs- und gerätespezifische Requesttransformation und –verarbeitung. Bereits unterstützte Anwendungsfelder sind Multichannel Publishing, welches eine generische Zugriffssteuerung auf multimediale Dokumentenstrukturen unter Nutzung medienspezifischer Manipulationsfunktionen wie Konvertierung und Komprimierung zur kontextabhängigen Darstellung und Präsentation multimedialer Dokumente ermöglicht, sowie wireless-orientiertes OLAP, mit dem eine Analyse über multidimensionale Datenstrukturen unterstützt wird.

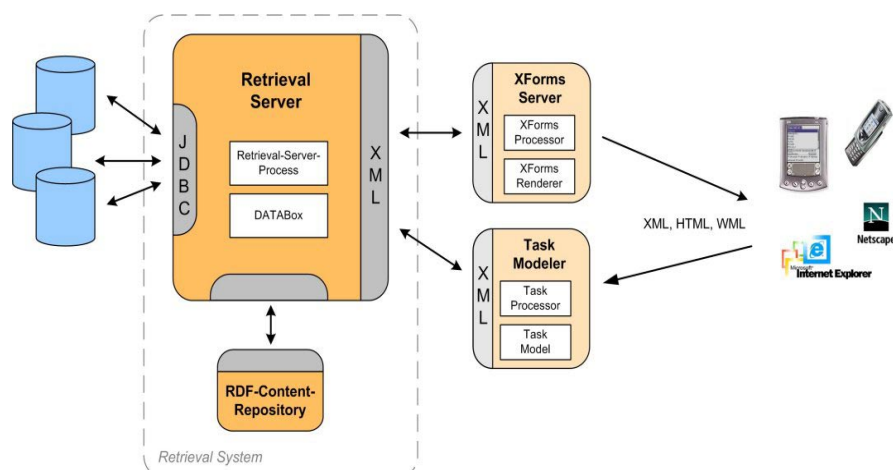


Abbildung 5-12: XPEA-Framework Architektur

<sup>1</sup> XPEA: XML-basierte Plattform für E-Business- und Content-Management-Applikationen

In Abbildung 5-12 ist der allgemeine Aufbau der XPEA-Architektur veranschaulicht. Hauptbestandteil der Architektur ist das Retrieval-System. Dieses realisiert die anwendungs- und gerätespezifische Anfragetransformation und -verarbeitung. Die Umsetzung der Aufgaben des Retrievalsystems erfolgt metadatenbasiert.

Bestandteile der Architektur des RetrievalSystems sind der Retrieval-Server und das RDF-ContentRepository. Der Retrievalserver hat Zugriff auf verschiedene integrierte Datenquellen und stellt für den Client eine XML-Schnittstelle zur Verfügung. Die Bearbeitung von durch den Client gestellte Anfragen erfolgt abhängig vom Anwendungskontext und den Geräteeigenschaften des Clients im Retrieval-Server-Process. Die Steuerung dieses Prozesses erfolgt über Metadaten, die durch ein RDF-Modell beschrieben im RDF-Content-Repository abgelegt sind. Der Metadatenbestand des Systems unterteilt sich in drei wesentliche Gruppen: Der *DeviceContext* enthält Informationen zu den Eigenschaften verschiedener Endgeräte, so unter anderem zu Graphikfähigkeiten und Rechenleistung. Der *DatabaseContext* beinhaltet Informationen zu den unterschiedlichen Datenbankmanagementsystemen (z.B. Multimediafähigkeiten) und zu den integrierten Datenbanken (z.B. Relationen, Attribute). Mit dem *Application Context* werden spezielle Sichten, welche die unterschiedlichen Anwendungen auf den Datenbestand haben, beschrieben. Eine solche Sicht beinhaltet z.B. für die Ergebnisdarstellung notwendige Attribute.

Zur Einbindung eines personalisierten Content Managements auf MPEG-7 beschriebenen audiovisuelle Daten muss das Retrievalsystem erweitert werden. Die notwendigen Erweiterungen und deren Umsetzung werden im nächsten Abschnitt dargelegt. Zunächst werden die allgemeinen Schritte der Requestbearbeitung durch den Retrievalserver erläutert.

### **Allgemeine Schritte der Requestverarbeitung**

Ziel der Requestverarbeitung ist es, basierend auf dem Metadatenbestand, für unterschiedliche Geräte und Anwendungen kontextabhängige Ergebnisse zu erstellen. Ist ein Nutzer an Dokumenten zu einem bestimmten Thema (einer speziellen Anwendung) interessiert, so sendet der Client eine Anfrage (Request) in XML-Syntax an den Retrieval Server. Im Request enthalten sind Angaben zum Endgerätetyp und zur Art der Anwendung sowie spezielle Anfrageparameter der Anwendung enthalten. Die Anfrage wird vom Server

validiert und analysiert. Die Abbildung 5-13 veranschaulicht den weiteren Ablauf des Retrieval Server Process.

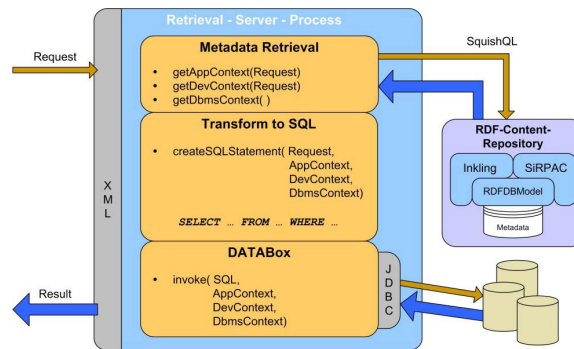


Abbildung 5-13: Allgemeiner Retrieval Server Process

Ausgehend von den im Request enthaltenen Informationen zur Art der Anwendung und des Clients wird ein Retrieval auf dem Metadatenbestand durchgeführt. Hierbei kommt eine in [Aude02] vorgestellte Erweiterung der Anfragesprache SquishQL zum Einsatz. Mit den aus dem Metadatenretrieval gewonnenen Informationen (Device Context, Application Context und Database Context) sowie den zusätzlichen Anfrageparametern, die in den Application Context eingebunden werden, kann eine SQL-Anfrage generiert werden. Diese wird an die entsprechende integrierte Datenbank weitergeleitet. Aus dem Ergebnis der Datenbankanfrage wird innerhalb der DATABox ein XML-Dokument generiert, welches als Ergebnis an den Client übermittelt wird.

## 5.2.2 Erweiterungen

Zur Durchführung eines personalisierten Content Managements auf MPEG-7 konformen Inhaltsbeschreibungen multimedialer Dokumente muss die eben beschriebene Architektur um eine Personalisierungskomponente erweitert werden. Zusätzlich muss eine geeignete RDF-Beschreibung der Anwendungssichten (Application Context) auf die MPEG-7 Daten gefunden werden. Dabei ist zu berücksichtigen, dass die MPEG-7 Daten auf unterschiedliche Arten verwaltet werden können. Nachfolgend wird die entwickelte Sichtdefinition vorgestellt, bevor die zusätzlich notwendigen Schritte zur Requestbearbeitung erläutert werden.

### Definition einer Sicht auf MPEG-7 Daten mittels RDF

Mit dem Applikationskontext wird eine Sicht auf für die Anwendung relevante Daten bereitgestellt. Die Beschreibung der Sicht erfolgt in einem RDF-Modell. Dabei werden die MPEG-7 Strukturen direkt, also unabhängig von der Art ihrer Verwaltung, unterstützt. Die Sicht auf Datenbestand wird als Sicht auf MPEG-7 Teilbäume hier auf eine Menge audiovisueller Segmente definiert. Es werden die aufgrund der Anforderungen der Anwendung für das Retrieval relevanten Teilelemente des MPEG-7 Baumes beschrieben. Für das Retrieval relevant sind Teilelemente, wenn sie für die Darstellung der Ergebnismenge notwendig sind (projection) oder wenn das Segment aufgrund bestimmter Eigenschaften des Teilelementes in die Ergebnismenge aufgenommen werden soll (selection). Abbildung 5-14 zeigt die schematische Darstellung einer RDF-Sichtdefinition auf MPEG-7 Teilbäume.

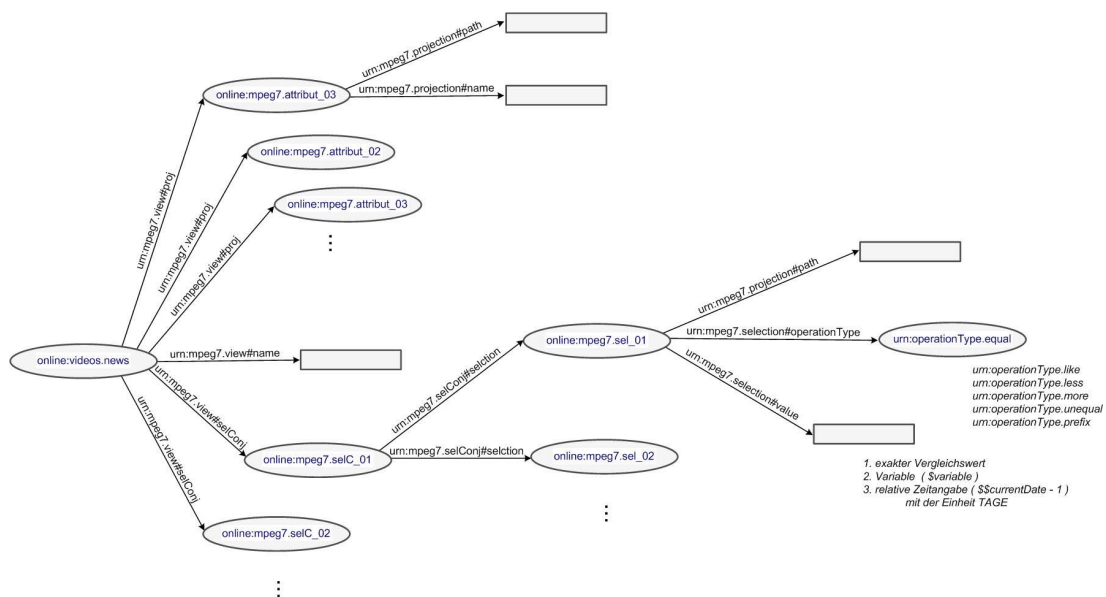


Abbildung 5-14: RDF-Sichtdefinition auf MPEG-7 Daten

Eine Sicht besteht aus einer Menge von Projektionen und einer Menge von Selektionen auf MPEG-7 Teilbäume. Die Projektionen sind durch die Pfadangabe des zu projizierenden Elemente im MPEG-7 Baum beschrieben. Diese erfolgt in X-Path-Notation. Als Wurzelement wird im konkreten Fall das eine gesamte Sendung repräsentierende Element

AudioVisualSegment genutzt. Der Pfad zum Titel einer Sendung ist damit mit *AudioVisualSegment/CreationInformation/Creation/Title/text()*, der zum Titel eines Beitrages mit *AudioVisualSegment/TemporalDecomposition7AudioVisualSegment/CreationInformation/Creation/Title/text()* anzugeben (vgl. Abschnitt 5.1.2). Neben der Pfadangabe zum zu projizierenden MPEG-7 Element besitzen Projektionen einen Namen, der von der Anwendung zur Ergebnispräsentation genutzt werden kann.

Selektionen sind wie Projektionen durch den Pfad des Selektionselementes und zusätzlich durch eine Vergleichsoperator und einen Vergleichswert, welche die Selektionsbedingung bilden, beschrieben. Mögliche Vergleichsoperatoren sind *equal* und *unequal* für den Vergleich auf Übereinstimmung des Elementes mit dem Selektionswert, *like* für den Test auf Vorhandensein des Selektionswertes innerhalb des Elementes sowie *more* und *less* für Vergleiche numerischer Elemente. Als Vergleichswerte können feste Werte, durch Nutzereingaben bestimmte variable Werte sowie durch das System bestimmte variable Werte angegeben werden. Die festen Werte sind durch die Anwendung bestimmt. In vorliegenden Anwendungsfall soll die Suche nur auf Nachrichtensendungen stattfinden. Ein fester Wert für eine Selektion ist in diesem Fall „news“ für das Element Genre. Die durch den Nutzer festgelegten variablen Werte werden zur Unterscheidung von festen Werten mit vorangestelltem \$, von System bestimmte variable Werte (in der konkreten Anwendung das aktuelle Datum) mit \$\$ gekennzeichnet. Nach erfolgtem Metadatenretrieval können die variablen Werte durch die Werte der Nutzereingaben, die als Anfrageparameter übergeben wurden, ersetzt werden.

Bei der Sichtdefinition auf MPEG-7 Daten sind Selektionen in konjunktiver Normalform darstellbar. Selektionen, die Alternativen ausdrücken sollen, bilden eine Konjunktion (*selConj*). Die Selektionen innerhalb einer solchen Konjunktion werden als durch ein logisches ODER verknüpft betrachtet, die Konjunktionen selbst als durch UND verknüpft.

Das Ergebnis einer RDF-Anfrage an eine auf diesem Schema basierende Sicht liefert als Applikation Context eine Menge von Projektionen und eine Menge von Selektionen<sup>3</sup> auf MPEG-7 Teilbäume. In dieser Arbeit werden diese Projektionen und Selektionen genutzt um eine SQL-Anweisung zu generieren, da die MPEG-7 Daten in einer relationalen Datenbank gehalten werden. Die Speicherung und Verwaltung von XML-Daten kann jedoch wie in Abschnitt 5.1.4 gezeigt auch auf andere Arten erfolgen. Die in dieser Arbeit entwickelte Schnittstelle kann dort ebenso genutzt werden, da die MPEG-7 Strukturen über die

Pfadangaben direkt in das RDF-Modell abgebildet werden. Die Selektionen und Projektionen auf den MPEG-7 Baum sind dann auf die jeweilige Speicherungsform abzubilden.

### **Zusätzliche Schritte bei Requestverarbeitung**

Mit der RDF-Sichtdefinition auf MPEG-7 Beschreibungen wurde die Voraussetzung für eine metadatenbasierte anwendungsspezifische Anfragetransformation und –verarbeitung geschaffen. Für eine personalisierte, auf dem content based Ansatz beruhende, Requestverarbeitung, sind neben den oben erwähnten allgemeinen Schritten Erweiterungen vorzunehmen. Die Nutzerinteressen müssen in die Requestverarbeitung einfließen. Dazu wird ein Nutzerkontext ermittelt, durch den die Verarbeitung zusätzlich gesteuert wird, indem die Sicht auf den Datenbestand erweitert und die Ergebnisse anhand der Interessen gefiltert werden.

Die Beschreibung der Anwendungssicht auf den MPEG-7 Datenbestand erfolgt unabhängig von der Art der Speicherung und des Zugriffs auf die Daten. In einer zusätzlichen Komponente muss die Abbildung auf die konkrete Speicherungsform, hier die in einer relationalen Datenbank, vorgenommen werden.

Um eine Personalisierung mittels inhaltsbasierte Filterung auf Grundlage von Nutzerpräferenzen durchzuführen, wird als dritte Erweiterung eine Rankingkomponente in das System eingefügt. Die Erweiterungen werden nun detailliert vorgestellt.

### **Ermittlung des Nutzerkontextes**

Die Suche auf audiovisuellen Dokumenten soll durch in den Nutzerprofilen abgelegte Nutzerinteressen durchgeführt oder unterstützt werden. Wie in Abschnitt 5.1.3 dargelegt können zwei Arten von Nutzerinteressen unterstützt werden:

**Bedingungen:** der Nutzer interessiert sich nur für Dokumente bekommen, die in einer bestimmten Eigenschaft einen festen Wert haben (z.B. nur für Nachrichtenbeiträgen mit den Genre „Sport“)

---

<sup>3</sup> in Konjunktiver Normalform

**Präferenzen:** der Nutzer interessiert sich für verschiedene Werte einer Eigenschaft unterschiedlich stark (z.B. ein höheres Interesse an „Sport“ als an „Politik“)

Die beiden Arten der Nutzerinteressen fließen auf unterschiedliche Weise in das Retrieval ein. Die Bedingungen stellen Selektionen auf dem Datenbestand dar. Der Nutzer möchte ausschließlich Dokumente erhalten, welche die Bedingungen erfüllen. Daher sind die Bedingungen schon bei der Anfragformulierung zu berücksichtigen. Sie werden als zusätzliche Selektionen (Konjunktionen) den Selektionen aus dem Applikationskontext hinzugefügt und fließen so schon in das boolesche Retrieval ein.

Dem Nutzer sollen die Ergebnisse nach Relevanz geordnete präsentiert werden. Dazu werden die im ersten Schritt des Retrievals erhaltenen Ergebnisse mit den Präferenzen verglichen und bewertet (content based filtering vgl. Abschnitt 2.2). Zu diesem Zweck müssen neben den für die Präsentation notwendigen Attributen (Projektionen des Applikationskontextes) auch die Attribute, über welche die Präferenzen beschrieben werden, Bestandteil der aus dem Retrieval erhaltenen Ergebnisse sein. Diese Attribute bilden zusätzliche Projektionen auf den Datenbestand, welche den Projektionen des Applikationskontextes hinzugefügt werden.

Die Ermittlung des Nutzerkontextes besteht also aus den Schritten (a) Ermittlung der Nutzerinteressen als Basis das Ranking der Anfrageergebnissen , (b) Ableitung zusätzlicher Selektionen aus den in den Interessen festgelegten Bedingungen (Gewicht 100 bzw. -100 vgl. Abschnitt 5.1.3) sowie (c) Ableitung von Projektionen aus den Präferenzen.

### **Generierung einer SQL-Anfrage an die MPEG-7 Datenbank**

In der vorliegenden Anwendung sind die MPEG-7-Beschreibungen der audiovisuellen Daten in einer relationalen Datenbank abgelegt. Für den Zugriff auf diese Daten muss aus den Selektionen und Projektionen auf den MPEG-7 Baum eine SQL-Anfrage generiert werden.

Dazu werden zunächst die Selektionen sowie der Projektionen aus Applikationskontext, Devicekontext und Nutzerkontext zu einer Menge von Projektionen und einer Menge von Selektionen zusammengefasst. Aus diesen Projektionen und Selektionen wird anschließend die SQL-Anweisung der Form „select ... from... where“ generiert. Dabei sind intern die zu den in den Projektionen und Selektionen enthaltenen Pfaden korrespondierenden Attribut- und Tabellennamen sowie notwendige Join-Bedingungen entsprechend dem konkreten Datenbankschema abgelegt. Dies geschieht in Form von Property-Value-Paaren. Den Pfaden

aus der Sichtbeschreibung (siehe oben) sind die korrespondierenden Datenbankattribute und Tabellennamen, den Tabellennamen notwendige Join-Bedingungen (siehe unten) zugeordnet.

Der select-Teil der SQL-Anweisung setzt sich aus den zu den in den Projektionen enthaltenden Pfaden korrespondierenden Datenbankattributen zusammen. Soll etwa der Titel eines Beitrages projiziert werden, also in der Ergebnisdarstellung enthalten sein, so muss „AudioVisualSegment.Title“ in die select-Liste aufgenommen werden.

Die from-Klausel enthält die Tabellennamen der Datenbankattribute der Projektions- und Selektionspfade. Soll eine Projektion auf den Titel eines Beitrages und eine Selektion über Keywords vorgenommen werden, so besteht die from-Klausel aus „AudioVisualSegment“ und „Keyword“.

Die where-Klausel setzt sich aus den Selektionsbedingungen der Selektionen und den notwendigen Join-Bedingungen zusammen. Die Selektionsbedingungen innerhalb einer Konjunktion selConj (siehe oben) werden dabei mittels *or* verknüpft, die Konjunktionen untereinander mittels *and*. Ein Beispiel für eine Selektion, die sich aus disjunktiv verknüpften Selektionsbedingungen zusammensetzt, ist die Selektion von Beiträgen die im Titel oder in den Keywords ein bestimmtes Wort, etwa „Olympia“, enthalten. Der Vergleichsoperator ist in diesem Fall *like* für das Vorhandensein innerhalb eines Elementes. Die aus den Pfadangaben abgeleiteten Datenbankattribute sind AudioVisualSegment.Title und Keyword.Keyword. Als Konjunktion für die where-Klausel ergibt sich hieraus „Keyword.Keyword like ‚Olympia‘ or AudioVisualSegment.Title like ‚Olympia‘“.

Als Basistabelle für die Joinbedingungen dient AudioVisualSegment (vgl. Abb. 5-10). Für alle anderen Tabellen müssen die Joinbedingungen mit AudioVisualSegment in die SQL-Anfrage aufgenommen werden. Um etwa audiovisuelle Segmente mit bestimmten Keywords zu selektieren, muss die Join-Bedingung „AudioVisualSegment.AudioVisualSegmentID = Keyword.AudioVisualSegmentID“ in die where-Klausel der SQL-Anfrage aufgenommen werden.

Zu unterstützen ist die Möglichkeit des Self-Joins der AudioVisualSegment-Tabelle. Sowohl die einzelnen Beiträge als auch die gesamte Nachrichtensendung sind audiovisuelle Segmente und in der Tabelle AudioVisualSegment abgelegt. Die Verknüpfung zwischen ihnen erfolgt über das Attribut TemporalDecompositioParent welches einem (Beitrags-) Segment das übergeordnete Elternsegment (Sendung) zuordnet. Um beim Self-Join eine



eindeutige Zuordnung der Attribute zu ermöglichen, muss einem Joinpartner ein neuer Name zugewiesen werden. Hier wird dem das Elternsegment repräsentierenden Teil der Name „main“ zugeordnet. Allen Pfade, die sich auf das übergeordnete Segment (die gesamte Sendung) beziehen, wird als Tabellename „AudioVisualSegment main“ und als Attributname „main.Attributname mAttributname“ zugeordnet. Ersteres bewirkt bei Aufnahme in die from-Klausel eine Umbenennung des Joinpartners, letzteres eine Umbenennung des zu projizierenden Attributes, was eine spätere Identifizierung im Resultset ermöglicht. Die Joinbedingung für die Tabelle „main“ lautet „main.AudioVisualSegmentId = AudioVisualSegment.TemporalDecompositionParent“.

Sind alle Join-Bedingungen ermittelt, so werden sie mittels *and* ,untereinander und mit den Selektionsbedingungen, zur where-Klausel zusammengefasst. Anschließend werden select-Klausel, from-Klausel und where-Klausel zur SQL-Anfrage zusammengefasst. Eine aus Projektionen auf den Titel eines Beitrages und den Titel der Sendung des Beitrages sowie aus der Selektion von Beiträgen die in den Keywords oder im Titel den Begriff „Olympia“ generierte Anfrage lautet dann:

```
select AudiovisualSegment.Title, main.Title mTitle
from AudiovisualSegment, AudiovisualSegment main, Keyword
where „main.AudiovisualSegmentId = AudiovisualSegment.TemporalDecompositionParent
and AudiovisualSegment. AudiovisualSegmentID = Keyword.AudiovisualSegmentID
and (Keyword.Keyword like ‚Olympia‘ or AudiovisualSegment.Title like ‚Olympia‘)
```

### Personalisiertes Ranking der Ergebnisse

Mit dem content-based filtering sollen dem Nutzer als Ergebnis einer Anfrage Dokumente geliefert werden, die von ihm gewünschte Attributwerte aufweisen und unerwünschte Werte nicht enthalten. Ein einfaches boolesches Retrieval hat den Nachteil, dass je nach Selektivität der Bedingung, zu viele oder keine Ergebnisse geliefert werden. Durch den Einsatz von Präferenzen können zum einen anstatt einer leeren Ergebnismenge auch Ergebnisse geliefert werden, die nicht vollständig dem Nutzerinteresse entsprechen. Zum anderen ist durch die Wichtung der Präferenzen eine Wichtung der Ergebnisse möglich, so dass die für den Nutzer interessantesten Ergebnisse zuerst geliefert werden.

Die zu bewertenden Ergebnisse werden durch eine wenig selektive (SQL-) Anfrage erhalten. Selektionsbedingungen sind dabei nur von der Art der Anwendung (Selektionen des Application context) abhängig und eventuell durch die Einbeziehung von Nutzerinteressen in Form von Bedingungen verschärft. Die Wichtung der Dokumente erfolgt mit Hilfe des Vektorraummodells. Dabei werden Nutzerinteressen und Dokumente als Vektoren im n-

dimensionalen Raum betrachtet, wobei die Dimensionen durch die Anzahl möglichen Terme bestimmt wird. Terme sind die Werte der für das Retrieval relevanten Dokumenteigenschaften, im konkreten Anwendungsfall also die Werte für Sprecher, Genre und Sender des Beitrages. Um ein Ranking durchführen zu können, wird den Termen eine Wertigkeit (Gewicht) zugeordnet. Im klassischen Textretrieval wird dies durch die Häufigkeit des Auftretens des Terms (Wort) im Dokument bestimmt.

Hier wird den Termen in den Dokumenten kein Gewicht zugeordnet. Sie besitzen entweder den Wert 1, wenn sie im Dokument vorhanden sind, oder den Wert 0, wenn dies nicht der Fall ist. Die Dokumenteigenschaften Genre, Beitrag und Sender werden als gleichwertig betrachtet und können so in einem gemeinsamen Dokumentenvektor aufgenommen werden. Die Terme der Nutzerpräferenzen, die hier die Anfrage darstellen, werden anhand der angegebenen Präferenzwerte gewichtet. Die Relevanz eines Dokumentes in bezug auf die Nutzerpräferenzen wird über die Ähnlichkeit von Anfrage- und Dokumentvektor bestimmt. Zwei Vektoren sind umso ähnlicher je größer der Wert der Ähnlichkeitsfunktion ist. Ein höherer Ähnlichkeitswert spiegelt also eine höhere Relevanz des Dokumentes in bezug auf die Anfrage wider. Am meisten genutzte Ähnlichkeitsfunktionen sind das Skalarprodukt und das Cosinusmaß. In dieser Arbeit wird das Cosinusmaß verwendet. In Tabelle 5-2 wird das Ranking beispielhaft demonstriert.

Term	pref	seg1	seg2	seg3	seg4	seg5
Genre = Sport	60	0	1	1	0	0
Genre = Außenpolitik	-40	1	0	0	1	0
Genre = Wetter	20	0	0	0	0	1
Sender = ARD	80	0	1	0	1	0
Sender = n-tv	-20	1	0	1	0	1
Sprecher = Bichmann	40	1	0	1	0	1
Sprecher = Brauner	60	0	1	0	0	0
Sprecher = Riewa	-20	0	0	0	1	0
<b>Skalarprodukt</b>		-20	200	80	20	40
<b>Cosinusmaß</b>		-0,09	0,86	0,34	0,09	0,17

**Tabelle 5-2:** Beispiel Ranking

Der Anfragevektor ist in der Spalte pref abzulesen. Er repräsentiert die Nutzerpräferenzen in bezug auf die einzelnen Terme. Die Spalten seg1 bis seg5 stellen die Dokumentenvektoren für Beispieldokumente dar. So ist das Dokument seg1 ein Außenpolitik-Beitrag des Senders n-tv mit der Sprecherin Bichmann.

Nach dem Ranking kann eine Filterung auf unterschiedliche Weise vorgenommen werden. Es können die n am höchsten bewerteten Dokumente oder alle Dokumente, die einen

Rankwert größer als ein bestimmter Schwellenwerte haben, als Ergebnisse geliefert werden. Die gefilterte Ergebnismenge sollte allerdings möglichst alle relevanten Ergebnisse (hoher recall-Wert) und nur relevante Ergebnisse (hoher precision-Wert) enthalten. Aus diesem Grund wird hier der Schwellenwert 0 zur Filterung genutzt. Alle Dokumente mit einem geringeren Rankwert sind für den Nutzer uninteressant., da ein negativer Präferenzwert ein Desinteresse ausdrückt. Im Beispiel würde das Segment seg1 nicht in die Ergebnismenge aufgenommen werden.

### Zusammenfassung Requestarbeit

Aus den eben vorgestellten zusätzlichen und den vorher beschriebenen allgemeinen Schritten zur Requestarbeit ergibt sich folgende, in Abbildung 5-15 dargestellte, Requestarbeit für die personalisierte Suche auf MPEG-7 beschriebenen Dokumenten.

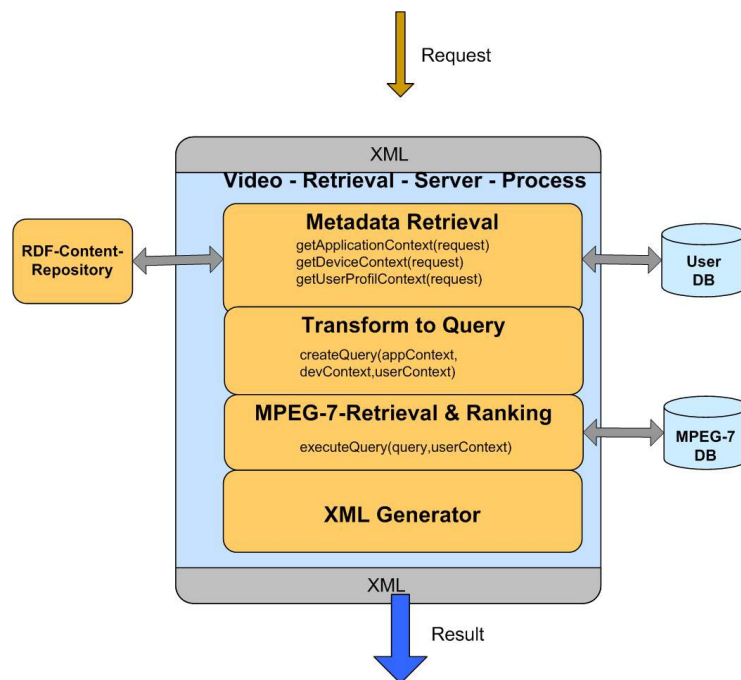


Abbildung 5-15: erweiterter Retrieval Server Prozess

Die *Metadata Retrieval*-Komponente wurde dabei um eine Funktion zur Bestimmung des Nutzerkontextes erweitert. In der *Transform to Query*-Komponente wird aus den im Metadata Retrieval ermittelten Projektionen und Selektionen eine der Speicherungsform der MPEG-7 Daten entsprechende Anfrage erzeugt. Im konkreten Fall ist dies eine SQL-Anfrage an eine relationale Datenbank. Durch die Repräsentation der Anfrage in Form von

Projektionen und Selektionen auf den MPEG-7 Baum können jedoch auch andere Formen der Speicherung und des Zugriffes auf XML-Dokumente unterstützt werden.

Die erzeugte Anfrage wird in der *MPEG-7 Retrieval & Ranking* –Komponente ausgeführt, wobei die Ergebnisse anhand der Nutzerpräferenzen aus dem `userContext`, wie oben beschrieben, gewichtet werden. Durch den modularen Aufbau der Serverkomponente können auch hier andere Komponenten zur Personalisierung eingesetzt werden. Ein Beispiel wäre das kommerzielle Produkt *Preference SQL [Pref]*. Es nutzt zur Modellierung von Nutzerinteressen ebenfalls Präferenzen. Allerdings werden keine Präferenzwerte angegeben sondern lediglich eine Rangfolge der Interessen in der Form „ich mag A mehr als B“. Ein Ranking und die Bestimmung eines Schwellenwertes finden dann auf Basis der Ebenen dieser relativen Ordnung statt. *Preference SQL* kann über ODBC und JDBC mittels eigener Treiber auf Standarddatenbanken zugreifen, so dass es in die vorgestellte Architektur integriert werden könnte.

Aus dem in der *MPEG-7 Retrieval & Ranking* gewonnenen gewichteten und gefilterten Ergebnis wird im *XML-Generator* ein zur Result-DTD konformes XML-Dokument erzeugt, welches dem Client übermittelt wird. Dort kann es mittels XSLT transformiert werden.

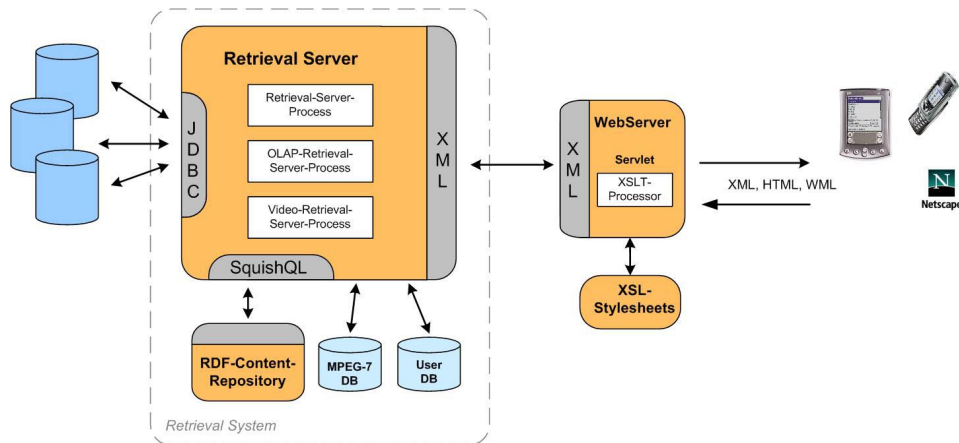
## 5.3 Prototypische Implementierung pViReSMo

Die in den vorangegangenen Abschnitten vorgestellten Konzepte wurden innerhalb der prototypischen Implementierung pViReSMo (personalized Video Retrieval Server Modul) umgesetzt. Dabei wurde auf die im Abschnitt 5.2 beschriebene XPEA-Architektur aufgesetzt und die vorgestellten Erweiterungen vorgenommen.

### 5.3.1 Überblick

Da sich die *taksmodelling*-Komponente der XPEA-Architektur (siehe Abb. 5-11) noch in der Entwicklung befindet wurde als Komponente zwischen Endgerät und Retrievalserver ein Jakarta Tomcat Webserver genutzt, auf dem ein Servlet die (http-) Anfragen der Clients in einen XML-Request wandelt und über eine Socketverbindung zum Retrievalserver weiterleitet. Damit ergibt sich die in Abbildung 5-16 dargestellte erweiterte XPEA-

Architektur die den personalisierten Zugriff auf MPEG-7 beschriebene Dokumente ermöglicht.



**Abbildung 5-16:** konkrete Frameworkarchitektur

Für den Retrievalserver fungiert der Webserver als Client, mit dem über ein XML-Schnittstelle kommuniziert wird. Der XML-Request wird dabei vom Video-retrieval-Server-Process abgearbeitet. Dabei wird über SquishQL auf das RDF-Content-Repository und über JDBC auf die angegliederte MPEG-7-Datenbank sowie die Nutzerdatenbank zugegriffen. Als Ergebnis der Requestbearbeitung wird ein XML-Dokument an den Webserver übermittelt. Dieses wird vom Servlet über einen XSLT-Prozessor auf Basis einer XSL-Stylesheets in eine vom Client darstellbare Form umgewandelt. Im konkreten Fall wird ein HTML-Dokument zur Darstellung in einem Webbrowser generiert. Es sind jedoch auch Umwandlungen in andere auf XML-basierende Sprachen wie WML für den WAP-Handy Bereich oder cHTML (compact HTML) für i-mode Anwendungen möglich.

Bei der prototypischen Implementierung waren somit folgende Teilkomponenten zu entwickeln:

**webbasierte Nutzerschnittstelle** zur Realisierung der Nutzeranfragen und Darstellung der Ergebnisse

**Servlet auf Webserver** zur Umwandlung der http-Anfragen in ein xml-Request und der Resultate in eine vom Client darstellbare Form

**Retrievalserver** zur Requestbearbeitung

### **Deskribierung der Videos**

Die zu entwickelten Komponenten dienen der Umsetzung der Anfragekomponente des Content Managements auf MPEG-7 Daten (vgl. Abb.5-1). Die Entwicklung einer Deskribierungskomponente war nicht Aufgabe dieser Arbeit. Die notwendige Deskribierung der Videos erfolgte mit MovieTool. MovieTool unterstützt die gesamte Breite des MPEG-7 Standards und ermöglicht die manuelle Unterteilung eines Videos in einzelne Segmente. Dadurch konnten die Nachrichtensendungen leicht in einzelne Beiträge segmentiert werden.

Der Beschreibungsvorgang setzt allerdings umfangreiche Kenntnisse des MPEG-7 Standards voraus, da die zur Beschreibung genutzten Deskriptoren einzeln aus dem Gesamtbaum ausgewählt werden müssen. Deskribierungskomponenten, die für reduzierte Schemas entworfen wurden, sind komfortabler in der Anwendung, da Eingabemasken für die Beschreibung der einzelnen Eigenschaften zur Verfügung gestellt werden können. Ein Beispiel für eine solche Komponente ist das von Rust in [Rust02] entwickelte VIDETO. Es kann allerdings aufgrund der eingeschränkten Beschreibungsmenge, die unter anderem die Eigenschaft Genre nicht enthält, in dieser Arbeit nicht eingesetzt werden.

Die mit Hilfe von MovieTool generierten MPEG-7 Beschreibungen (XML-Dokumente) werden mit Hilfe eines Java-Parsers ( XML2DB- Komponente vgl. Abb. 5-1) eingelesen und anhand der in 5.1.4 vorgestellten Abbildungsvorschrift in die MPEG-7 Datenbank eingefügt.

### **Ermittlung der Nutzerpräferenzen**

Die Ermittlung der Nutzerinteressen war nicht Hauptbestandteil der Arbeit. Insbesondere wurde keine Konzepte zur Erleichterung der Eingabe der Interessen, was in konkreten Anwendungen zu einer Verbesserung der Qualität der Angaben führen kann, eingesetzt.

Die Nutzerpräferenzen können über ein Webinterface in einem einfachen Formular angegeben werden wobei Interessen die Art des Beitrage, den Sender des Beitrages, sowie den Sprecher des Beitrages betreffend mit einzelnen Wertigkeiten versehen werden können. Es wird also ein aktives Profiling durchgeführt. Passives Profiling wird mit dem entwickelten Prototyp nicht unterstützt. Es könnte jedoch durch eine Auswertung der Serverlogs, in denen der Zugriff protokolliert wird, umgesetzt werden. Durch den Charakter des vorliegenden Anwendungsfeldes mit sich aufgrund aktueller Ereignisse kurzfristig

ändernden Nutzerinteressen ist eine solche Auswertung sehr komplex. Die aus dem aktiven Profiling gewonnenen Nutzerinteressen werden direkt in die Nutzerdatenbank eingefügt. Eine Generierung von XML-Dokumenten für die Nutzerpräferenzen findet zur Zeit noch nicht statt. Dies wäre dann notwendig, wenn die Nutzerdaten ausgetauscht werden sollen.

### **5.3.2 Anfragekomponente**

Die in dieser Arbeit entwickelte Anfragekomponente bietet die Möglichkeit einer personalisierten Suche auf Nachrichtenbeiträgen in zwei Schritten. Im ersten Schritt wird dem Nutzer eine Liste ihn interessierender Beiträge geliefert. Im zweiten Schritt kann er einen Beitrag aus der Liste auswählen, der ihm dann präsentiert wird.

#### **Suche nach Beiträgen**

Die webbasierte Nutzerschnittstelle bietet die Möglichkeit zwei Arten von Anfragen zu stellen. Zum einen kann der Nutzer eine Anfrage absetzen, die ausschließlich auf Informationen aus seinem Nutzerprofil beruht. Hierbei wird in der Eingabemaske (Abbildung 5-17) keine weitere Auswahl vorgenommen.

Der Nutzer hat jedoch auch die Möglichkeit, zusätzliche Suchparameter anzugeben. In diesem Fall wird die Anfrage durch die Profildaten vervollständigt. Mögliche Suchparameter sind ein bestimmtes Nachrichtengebiet oder ein bestimmter Sender sowie die Aktualität des Nachrichtenbeitrages. Außerdem kann ein Suchbegriff angegeben werden, über den in den Titeln der Nachrichtenbeiträge sowie in Schlagwörtern, die den Beiträgen zugeordnet sind, gesucht wird.

Wählt der Nutzer in der Eingabemaske ein bestimmtes Ressort oder einen bestimmten Sender aus, so werden bei der Anfrage die Profildaten für die ausgewählte Eigenschaft nicht berücksichtigt. Sie werden bei der Ermittlung des Nutzerkontextes ausgeblendet. Der Nutzer hat so die Möglichkeit bei bestimmten Anlässen Ergebnisse zu erhalten, die seinen Profildaten nicht vollständig entsprechen, ohne die Profildaten explizit zu ändern. Dieses Ausblenden der Profildaten kann als kurzzeitiges Profilupdate verstanden werden. Es hat den gleichen Effekt wie das Setzen einer Bedingung mit diesem Attributwert.

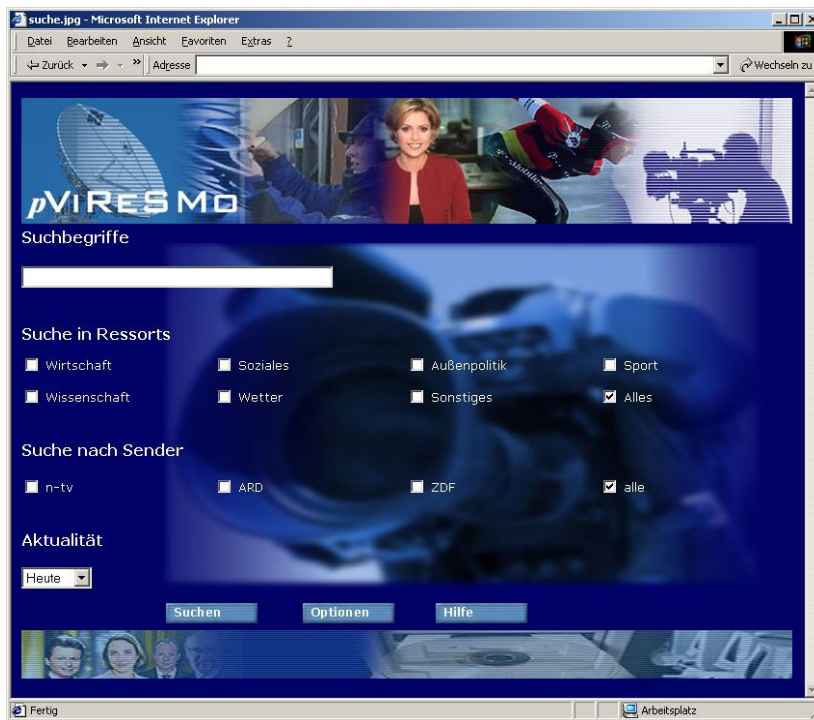


Abbildung 5-17: pViReSMo Webschnittstelle

Mit diesem „versteckten Profilupdate“ wird berücksichtigt, dass sich Nutzerinteressen im Nachrichtsbereich aufgrund von besonderen Ereignissen kurzfristig ändern können. So kann ein Interesse für Sport nur bei bestimmten Großereignissen vorhanden sein, während Sport sonst als uninteressant angesehen wird. Wird nun das Ressort „Sport“ ausgewählt, so werden die Präferenzen aus dem Nutzerprofil, welche die Eigenschaft „Art des Beitrages“ betreffen, nicht berücksichtigt und die sonst negativ bewerteten Beiträge zum Thema Sport als Ergebnis geliefert.

Die Daten des Eingabeformulars werden zum Webserver geschickt, auf dem ein Servlet aus den Parametern des http-request ein XML-Dokument generiert und dieses über eine Socketverbindung an den Retrieverserver weiterleitet. Beispiel 5-3 zeigt ein solches Request-Dokument.

```
<REQUEST DEVICE="PC">
  <SELECT userID="micha">
    <GENRE>all</GENRE>
    <QUERY>Olympia</QUERY>
    <DISTRIBUTOR>all</DISTRIBUTOR>
  </SELECT>
</REQUEST>
```

Beispiel 5-3: XML-Request zur Anfrage an den RetrievalServer



Ein Request besteht aus einem Element welches den Namen der Anwendung zur Bestimmung des Applikationskontextes repräsentiert. In diesem Fall ist dies *SELECT* für die Anwendung „Selektion relevanter Beiträge“. Es hat das Attribut *userID*, welches zur Bestimmung des Nutzerkontextes verwendet wird. Innerhalb des Elementes befinden sich weitere Elemente, die Parameter der Anwendung darstellen.

Der Retrievalserver arbeitet den Request wie in vorigen Abschnitt beschrieben ab. Das Ergebnis der Abarbeitung ist ein XML-Dokument welches die relevanten Resultate der Suche in der den Nutzerpräferenzen entsprechenden Rangfolge beinhaltet. Das Beispiel 5-4 zeigt einen Ausschnitt aus einem solchen XML-Dokument.

```
<Results>
  <RESULT rankvalue="0,53">
    <GIVENNAME>Wolf</GIVENNAME>
    <FAMILYNAME>von Lojewski</FAMILYNAME>>
    <DISTRIBUTOR>ZDF</DISTRIBUTOR>
    <REDUCEDFREETEXTANNOTATION> Damenstaffel gewinnt Gold ...
  </REDUCEDFREETEXTANNOTATION>
    <AUDIOVISUALSEGMENTID>14</AUDIOVISUALSEGMENTID>
    <TITLE>Gold für Langlaufstaffel</TITLE>
    <MTITLE>ZDF heute Journal</MTITLE>
    <RELEASE>2002-02-21 21:45:00</RELEASE>
    <MEDIAINCRDURATION>2640</MEDIAINCRDURATION>
  </RESULT>
  <RESULT rankvalue="0,43">
    ...
  </RESULT>
  ...
</Results>
```

#### Beispiel 5-4: XML-Resultat einer Anfrage

Ein Resultat besteht dabei aus den zur Darstellung des Ergebnisses notwendigen Attributen, die als Projektionen des Applikationskontext im RDF-Repository ermittelt wurden. Dazu zählen der Titel, Dauer, Sprecher, Sender und eine textuelle Zusammenfassung des Beitrages sowie die Sendung aus der der Beitrag stammt und die ID des Beitragsegments.

Das Ergebnisdokument wird an den Webserver übertragen. Dort wandelt das Servlet das XML-Dokument mittels eines XSL-Stylesheet in ein HTML-Dokument um. Dieses wird dann an den Client gesendet. Das Ergebnis ist eine geordnete Liste der relevanten Nachrichtenbeiträge. Die Abbildung 5-18 zeigt eine solche Ergebnisliste.

Die zu den einzelnen Nachrichtenbeiträgen angezeigten Screenshots sind lokal auf dem Webserver abgelegt. Die URL dieser Bilder wird in dieser prototypischen Implementierung durch das Stylesheet über die SegmentID des Nachrichtenbeitrages generiert. Sie kann in einer weiterführenden Umsetzung auch innerhalb der MPEG-7 Beschreibung des Videos als Related-Material-Eigenschaft eines Segmentes angegeben und im XML-Resultat mitgeliefert werden. Dazu ist lediglich eine Erweiterung des MPEG-7 Schema um eben dieses Related-

Material-Element und der Applikationskontext um die Projektion auf dieses Element notwendig.

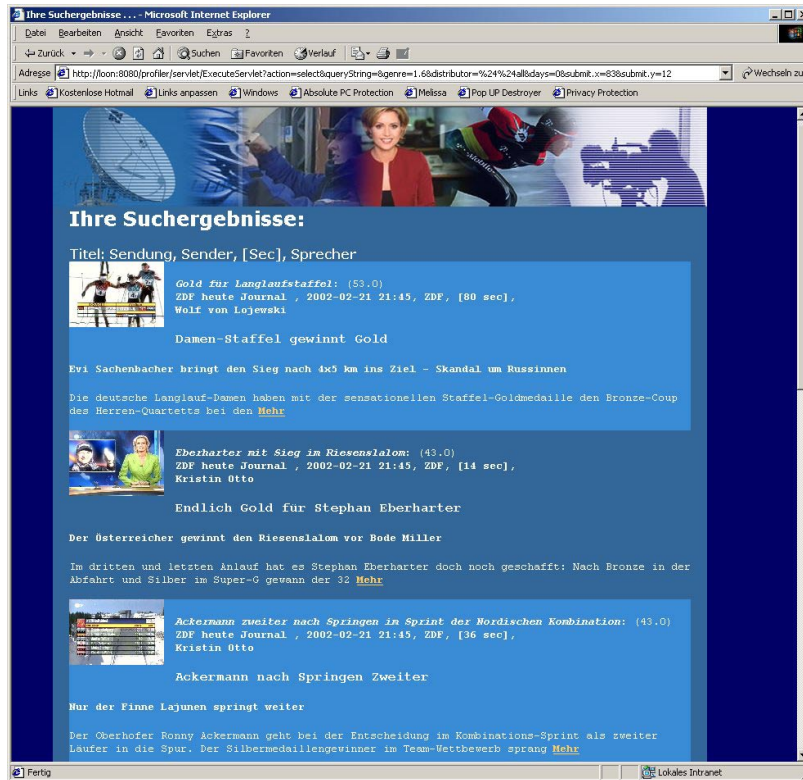


Abbildung 5-18: Ergebnisliste

Der Nutzer kann nun den Beitrag aus der Ergebnisliste auswählen, den er sich ansehen will, indem er auf das zugehörige Bild klickt. Damit wird wiederum ein http-request an das Servlet gesendet, welcher dort in ein XML-Request gewandelt und zum Server übermittelt wird. Ein solcher Request ist in Beispiel 5-5 angegeben.

```
<REQUEST DEVICE="PC">
  <FetchVideo userID="micha">
    <SEGMENTID>14</SEGMENTID>
  </FetchVideo>
</REQUEST>
```

#### Beispiel 5-5: XML-Request für Beitragspräsentation

Der Name der Anwendung, der für das Metadatenretrieval, zur Ermittlung des Applikationskontextes genutzt wird, ist dabei FetchVideo. Der einzige notwendige Anwendungsparameter ist die *SegmentID*. Das Attribut *userID* ist mit enthalten, um ein

passives Profiling zu ermöglichen, was in der derzeitigen Implementierung jedoch noch nicht unterstützt wird.

Das Resultat der Requestbearbeitung ist ein XML-Dokument, das dem aus Beispiel 5-4 ähnelt. Es enthält ein Result-Element, welches, wie im Application Context für die Anwendung FetchVideo festgelegt, die zur Darstellung des Videos notwendigen Elemente MediaRelTimePoint (Startpunkt des Beitrages innerhalb des Videos), MediaIncrDuration (Dauer des Beitrages) und MediaURI (zur Lokalisation des Videos) enthält.

### Darstellung eines Beitrages

Das XML-Ergebnisdokument wird vom Servlet über ein XSL-Stylesheet in ein SMIL-Dokument umgewandelt. SMIL (Synchronized Multimedia Integration Language) [Hosc98] ist eine vom W3C als Standard vorgeschlagenen auf XML basierende Sprache zur Steuerung webbasierter multimedialer Präsentationen. Die Präsentation kann dabei alle Arten von Multimediaobjekttypen enthalten. In einem SMIL-Dokument können Startpunkte und Dauer der Wiedergabe von Audio und Videodokumenten festgelegt werden. Dies wird hier ausgenutzt, um den Teil des Videos zu präsentieren, der den vom Nutzer gewünschten Nachrichtenbeitrag enthält. Das SMIL-Dokument für den ersten Beitrag der Ergebnisliste aus Abbildung 5-17 sieht wie folgt aus.

```
<smil>
  <head>
    <meta content="heute journal" name="title"/>
  </head>
  <body>
    <par>
      <video region="videoregion" src="file:///F:/6 zdf - heute journal
gesamt.mpg" clip-begin="17.910102" dur="80.077"/>
    </par>
  </body>
</smil>
```

Ein SMIL-Dokument beinhaltet das wiederzugebende Video in Form eines Links auf das Video. In der derzeitigen Anwendung sind die Videos als lokale MPEG-1 Files abgelegt. Die Videos könnten jedoch auch in anderen Formaten, so in Streaming-Formaten auf einem Videosever vorliegen.

Mit dem RealPlayer der Firma RealNetworks existiert eine weit verbreitete Software zur Wiedergabe von SMIL-Präsentationen, die im hier entwickelten Prototyp für die Darstellung der Beiträge eingesetzt wird. Er wird vom Webbrowser aufgerufen, wenn der

Typ des http-response auf „application-smil“ gesetzt ist. Diese Setzen des response-type geschieht durch das übermittelnde Servlet. Die Darstellung des Videobeitrages mit dem RealPlayer illustriert Abbildung 5-19.



Abbildung 5-19: Darstellung eines Nachrichtenbeitrages

Die Darstellung von Videosegmenten kann neben dem RealPlayer auch der über *Windows Media Files* gesteuerte Microsoft Mediaplayer erfolgen. Diese, unter anderem in [Rust01] genutzte, Lösung ist allerdings nur mit dem Microsoft Internet Explorer umzusetzen, und daher nicht auf allen Plattformen einsetzbar.

### 5.3.3 Fazit

Mit der Implementierung des Prototypen pViReSMo konnten die zuvor entwickelten Konzepte für ein personalisiertes universelles Content Management auf MPEG-7 beschriebenen audiovisuellen Daten innerhalb des vorgegebenen Anwendungsszenarios umgesetzt werden.

Es wurde eine webbasiert Anwendungsumgebung mit einem Webserver, der die XML-Schnittstelle zum RetrievalServer und eine http-Schnittstelle zum Client bereitstellt, gewählt. Innerhalb der Anfragekomponente wurde berücksichtigt, dass sich die Nutzerinteressen im Anwendungsszenario Nachrichten durch besondere Ereignisse kurzfristig ändern können. Spezielle Interessen können direkt angegeben werden, wobei

dann die langfristigen Interessen im gewählten Bereich während des Filterungsprozesses ausgeblendet werden.

# 6 Schlussbetrachtungen

In diesem Kapitel werden die Ergebnisse der Arbeit zusammengefasst und ein kurzer Ausblick auf mögliche Erweiterungen der hier vorgestellten Lösungen gegeben.

## 6.1 Zusammenfassung

Mit dieser Arbeit wurde die Konzeption und Umsetzung eines universellen personalisierten Content Managements audiovisueller Daten vorgelegt. Zunächst wurden verschiedene Personalisierungskonzepte, bezogen auf die Modellierung von Nutzerinteressen, Gewinnung von Nutzerinteressen und Informationsfilterung auf Basis der Nutzerinteressen, analysiert. Dabei wurde eine Anwendungsabhängigkeit des Einsatzes der einzelnen Ansätze festgestellt und eine passende Lösung für das gegebene Anwendungsszenario, die personalisierte Suche auf Nachrichtenbeiträge in Form von Videos, gewählt.

Als Filtermethode wird das content based filtering genutzt. Die notwendige Repräsentation von Objekteigenschaften und Nutzerinteressen erfolgt dabei mittels MPEG-7, einem neuen XML-basierten Metadatenstandard zur Beschreibung audiovisueller Daten mit dem, neben Eigenschaften audiovisueller Daten auch Nutzerinteressen und –verhalten diese Daten betreffend beschreibbar sind. Die umfangreichen Beschreibungsmöglichkeiten erlauben die Umsetzung der verschiedenen Personalisierungskonzepte auf Basis standardisierter Nutzerinformationen.

Die in MPEG-7 gegebene Möglichkeit zur Angabe von Wertigkeiten einzelner Nutzerinteressen konnte genutzt werden, um Nutzerinteressen als Präferenzen so zu modellieren, dass die Bevorzugung eines Wertes einer Objekteigenschaft gegenüber eines anderen Wertes ausgedrückt werden konnte. Auf Basis der gewichteten Nutzerinteressen konnte zur Filterung relevanter Inhalte ein personalisiertes Ranking mittels Vektorraummodell durchgeführt werden, indem die Werte der einzelnen Präferenzen zur Wichtung des Anfragevektors genutzt wurden.

MPEG-7 stellt eine äußerst große Menge von Beschreibungselementen zur Verfügung. Da für die konkrete Anwendungssituation nur ein Teilmenge dieser Beschreibungselemente notwendig war, wurde das Schema sowohl für die Nutzerbeschreibungen als auch für die Beschreibungen der Daten auf die notwendigen Deskriptoren reduziert. Zur effektiven Verwaltung werden die MPEG-7 Daten in einer relationalen Datenbank gehalten. Durch die Reduzierung des MPEG-7 Schemas wurde die notwendige Abbildung der XML-Strukturen der MPEG-7 Beschreibungen auf relationale Strukturen ebenso wie die Erzeugung der Daten selbst erleichtert.

Für den universellen Zugriff erfolgt die Einbindung in das am ZGDV-Rostock entwickelte XPEA-Framework. Die Steuerung des personalisierten Content Management erfolgt dabei metadatenbasiert. Zur Integration von MPEG-7 Daten in das Framework erfolgte eine Kopplung vom MPEG-7 Beschreibungen mit dem RDF-Standard, indem mittels RDF eine Sicht auf MPEG-7 Daten definiert wurde. Diese Sicht besteht aus einer Menge von Projektionen und Selektionen auf den MPEG-7-Baum. Diese, sich an den Strukturen vom MPEG-7 Beschreibungen orientierende, Sichtdefinition ermöglicht einen universellen Einsatz unabhängig von der konkreten Art der Verwaltung der MPEG-7 Daten.

Der zur Durchführung einer personalisierten Informationsfilterung in das Framework integrierter VideoRetrievalServer zeichnet sich durch seinen modularen Aufbau aus. Dadurch bleibt er erweiterbar und leicht an veränderte Beschreibungsschemata, MPEG-7 Verwaltungsarten und Personalisierungskonzepte anpassbar.

In einer prototypischen Implementierung konnten die entwickelten Konzepte im gegebenen Anwendungsszenario erfolgreich umgesetzt werden.

## 6.2 Ausblick

Im folgenden wird auf einige Aspekte eingegangen, die in dieser Arbeit nicht ausreichend berücksichtigt werden konnten. Außerdem werden Möglichkeiten zur Erweiterung des Prototypen genannt.

Datenschutzaspekte wurden in dieser Arbeit nicht in Betracht gezogen. Sie sind bei der zentralen Speicherung privater Informationen aber nicht immer zu vernachlässigen. Dabei spielt die Art der gespeicherten Information eine Rolle. Sicherlich ist die Speicherung von Interessen weniger kritisch zu sehen als die persönlicher „privater“ Informationen, wie Name

oder Adresse. Eine Anonymisierung von Nutzernamen erschwert die Zuordnung von Interessen zu konkreten Personen. Weitere Konzepte wären zu untersuchen

Dies gilt auch für den Bereich der Sicherung von Urheberrechten. Für beide Bereiche stellt der MPEG-7 Standard Beschreibungsmöglichkeiten zur Verfügung. Die Umsetzung und Auswahl geeigneter Konzepte ist jedoch von der jeweiligen Anwendung zu leisten

Eine Erweiterungsmöglichkeit des Prototypen liegt in dem Ausbau der Komponente für das aktive Profiling. In dieser Arbeit lag das Hauptaugenmerk auf der Verwendung der Profilinformatoren, so dass auf eine nutzerfreundliche Umsetzung, die mit einfachen übersichtlichen Mitteln vollständige Nutzerangaben ermitteln kann, zugunsten anderer Arbeiten verzichtet wurde.

Auch eine Einbeziehung mobiler Geräte wurde noch nicht vorgenommen. Die Voraussetzungen dafür sind jedoch schon gegeben, da das Schema die Beschreibung unterschiedlichen Versionen eines Videos unterstützt und das Content-Repository leicht um Informationen zum DeviceContext erweitert werden kann.



# Literaturverzeichnis

- [Aude02] Audersch, Stefan: Universelle Metadatenverwaltung zur Steuerung von Transformations- und Daten-Analyse-Prozessen. 14.GI-Workshop Grundlagen von Datenbanken, Darß 2002
- [Baud98] Baudisch, Patrick: Recommending TV Programs: How Far Can We Get at Zero User Effort? in: Recommender Systems. Papers from the 1998 Workshop: AAAI Press, Technical Report WS-98-108. (1998)
- [Baud99] Baudisch, Patrick: Joining Collaborative and Content Based Filtering. in Online Proceedings of the CHI '99Workshop on Interacting with Recommender Systems, 1999 siehe: <http://www.darmstadt.gmd.de/rec99>
- [Baud01] Baudisch, Patrick: Dynamic Information Filtering. Dissertation an der technischen Universität Darmstadt. (2001)
- [BCZS01] Benitez, Ana B., Di Zhong, Shih- Fu Chang and John R. Smith: MPEG-7 MDS Content Description Tools and Applications. in: W. Starbek (Ed.): CAIP2001, LNCS 2124, pp 41-52, Springer-Verlag Berlin Heidelberg, 2001
- [BeCr92] Belkin, Nicholas J. und W. Bruce Croft: Information Filtering and Information Retrieval: Two Sides of the Same Coin?: Communications of the ACM, 35 (12), 1992
- [Bour02] Bourret, Ronald: XML and Databases, 2002 siehe <http://www.rpbourret.com/xml/XMLAndDatabases.htm>
- [BrGu00] Brickley, Dan und R.V. Guha: Resource Description Framework (RDF) Schema Specification 1.0 Technical Report, W3C, 2000
- [Ditt01] Dittberner, Christoph: Zugriff auf multimediale XML-Dokumente in objekt-relationalen Datenbanken. Diplomarbeit an der Universität Rostock, 2001
- [FoDu92] Foltz, Peter W. and Susan T. Dumais: Personalized Information Delivery: An Analyses of Information Filtering Methods in: Communications of the ACM, 35(12) pp51-60, 1992

- [Gill00] Gilliland-Swetland, Anne J.: Introduction to Metadata: Setting the Stage. Technical Report, Getty.edu 2000
- [Gree98] Greening, Dan: Collaborative Filtering for WebMarketing Efforts. Recommender Systems. Papers from the 1998 Workshop. Menlo Park, CA 1998
- [HePa00] Heery, R. and M.Patel: Application Profiles: mixing and matching metadata schemas, Ariadne Issue 25, September 2000 siehe <http://www.ariadne.ac.uk/issue25/app-profiles>
- [HuIa98] Hunter, Jane and R. Iannella: The Application of Metadata Standards to Video Indexing. Second European Conference on Research and Advanced Technology for Digital Libraries, Crete, Greece, September, 1998
- [HuLa01] Hunter, Jane and Carl Lagoze: Combining RDF and XML Schemas to Encance Interoperability between Metadata Application Profiles. WWW10, Honkong, 2001
- [HuAr00] Hunter, Jane and Liz Armstrong: A Comparison of Schemas for Video Metadata Representation. Proceedings of the 8<sup>th</sup> International World Wide Web Conference, Toronto, Canada May 2000
- [KBW01] Kießling, Werner, Wolf-Tilo Balke and Mathias Wagner: Personalized Content Syndication in a Preference World. in: Proceedings of the EnCKompass Workshop on E-Content Management (EnCKompass 2001), pp. 3-6, Eindhoven, The Netherlands, June 2001.
- [KIMe00] Klettke, Meike und Holger Meyer: Manging XML-documents in object-relational databases. XML2000, 2000
- [KRBH98] Konstan, Joseph, John Riedle, Al Borchers and Jonathan Herlocker: Recommender systems: A GroupLense Perspective. Recommender Systems. Papers from the 1998 Workshop. Menlo Park, CA, 1998
- [LaSw99] Lassila, Ora R. und Ralph R. Swick: Resource Description Framework (RDF) Model and Syntax Specification W3C Recommendation, 1999
- [MPEG01a] MPEG MDS Group: MPEG-7 Part 5 Multimedia Description Scheme (MDS). ISO, 2001

- [MPEG01b] MPEG MDS Group: Information technology – Multimedia Framework (MPEG-21) Part 3: Digital Item Identification and Description. ISO, 2001
- [MPEG01c] MPEG Requirements Group: MPEG-7 Profiles Under Consideration. ISO, 2001
- [MPEG01d] MPEG MDS Group: Information technology – Multimedia Framework (MPEG-21) Part 1: Visions, Technologies and Strategy. ISO, 2001
- [MPEG01e] MPEG Requirements Group: MPEG-21 Use Case Scenarios. ISO, 2001
- [MPR00] Manber, Udi, Ash Patel und John Robinson: Experience with Personalization on Yahoo!. in: Communications of the ACM Vol.43, No.8 August 2000
- [Pref] Preference SQL 1.2. Technical White Paper, Database Preference Software GmbH, 1999
- [RaSa01] Ramakrishnan, Naren und Saverio Perugini: The Partial Evaluation Approach to Information Personalization. ACM Transactions on Information Systems, August 2001
- [RaSm01] Rafter, Rachael und Barry Smyth: Passive Profiling from Server Logs in an Online Recruitment Environment. in: Proceedings of the IJCAI Workshop on Intelligent Techniques for Web Personalisation (ITWP 2001) Seattle, Washington, USA, 2001
- [Rehm00] Rehm, Eric: Representing Internet Streaming Media Metadata using MPEG-7. in Proceedings of the ACM-MM 2000. ACM, 2000
- [Riec00] Riecken, Doug: Personalized Views of Personalization. in: Communications of the ACM August 2000 Vol.43, No.8 (2000)
- [Rust01] Rust, Mathias: Multimedia Content Management für interaktive TV-Applikationen unter Einbeziehung des MPEG-7 Standards. Diplomarbeit an der Universität Rostock. 2001
- [SmCo99] Smyth, Barry und Paul Cotter: Surfing the Digital Wave: Generating Personalized Television Guides using Collaborative, Case-based Recommendation. in: Proceedings of the 3<sup>rd</sup> International Conference on case-based Reasoning, Munich, Germany, 1999

- [Hosc98] Hoschka, P: Synchronized Multimedia Integration Language, W3C Recommendation, 1998
- [STH+99] Shanmugasundaram, Jayavel, Kristin Trufte, Gang He, Chun Zhang David DeWitt and Jeffrey Naughton: Relational Databases for Querying XML Documents: Limitations and Opportunities. in. Proceedings of the 25<sup>th</sup> VLDB Conference, Edinburgh, Scotland, 1999
- [TVA] TV-Anytime Forum *<http://www.tv-anytime.org>*
- [URI01] Uniform Resource Identifiers (URI): Generic Syntax. *<http://www.ietf.org/rfc/rfc2396.txt>*
- [VaVa01] Varlamis, Iraklis und Michaelis Vazirgiannis: Bridging XML-Schema and relational databases. in: the proceedings of ACM Symposium on Document Engineering, Atlanta, USA, Nov. 2001
- [XMLS01] XML Schema Part 0: Primer. Technical Report W3C XML Schema Workgroup, World Wide Web Consortium, 2001 siehe *<http://www.w3.org/TR/xmlschema-0>*
- [Zeit01] Zeitz, Andre: Personalisierungskonzepte für Digitale Bibliotheken. Diplomarbeit an der Universität Rostock, 2001

# Thesen

1. Die große Menge der heute zur Verfügung stehenden audiovisuellen Daten erfordert ein personalisiertes Content Management. Für den personalisierten Zugriff auf Informationen existiert eine Reihe von Konzepten, welche die Modellierung der Nutzerinteressen, die Gewinnung von Nutzerinteressen sowie die Filterung relevanter Inhalte auf Basis von Nutzerinteressen zum Gegenstand haben. Sie sind allerdings nicht universell sondern anwendungsspezifisch einzusetzen.
2. Zur Umsetzung eines personalisierten Content Managements über audiovisuelle Daten sind Beschreibungen der Daten mittels Metadaten unverzichtbar. Mit MPEG-7 steht ein neuer, XML-basierter Metadatenstandard zur Verfügung, der eine umfassende Beschreibung audiovisueller Daten ermöglicht.
3. Neben audiovisuellen Daten lassen sich mit MPEG-7 auch diese Daten betreffende Interessen und Verhalten von Nutzern beschreiben. Diese Beschreibungen ermöglichen den Einsatz verschiedener Personalisierungskonzepte auf der Basis standardisierter Nutzerinformationen.
4. Der grosse Beschreibungsumfang vom MPEG-7 muss sowohl in Bezug auf die Daten als auch auf die Nutzerinformationen auf für die Anwendung notwendige Beschreibungselemente reduziert werden, um eine effiziente Deskribierung und Verwaltung der Daten zu ermöglichen.
5. Zur Zugriffsunterstützung ist die Verwaltung von MPEG-7-konformen Beschreibungen in Datenbanken notwendig. Die Abbildung von XML-Strukturen der MPEG-7 Beschreibungen auf relationale Datenbankstrukturen erfordert die Beachtung von Beziehungen zwischen den MPEG-7 Elementtypen.
6. Zur personalisierten, inhaltsbasierten Filterung relevanter Daten ist das Vektorraummodell geeignet. Durch eine Modellierung von Nutzerinteressen in Form von Präferenzen und die Angabe von Wertigkeiten für die einzelnen Präferenzen kann dabei der Anfragevektor gemäß den Nutzervorgaben gewichtet werden und so ein personalisiertes Ranking stattfinden.
7. Um einen universellen Zugriff auf MPEG-7 beschriebenen Daten zu ermöglichen, lässt sich die am ZGDV Rostock entwickelte XPEA-Architektur einsetzen. Eine in diesem Zusammenhang zu entwickelnde RDF-Sicht auf MPEG-7 beschriebene Daten, muss die verschiedenen Arten der Verwaltung von MPEG-7 Beschreibungen berücksichtigen. Dies kann durch eine universelle Beschreibung der Anwendungssicht als Menge von Selektionen und Projektionen auf einen MPEG-7 Baum geschehen.
8. Durch einen modularen Aufbau bleibt die Video-Retrievalkomponente des XPEA-Frameworks erweiterbar und leicht an veränderte Beschreibungsschemas, MPEG-7 Verwaltungsarten und Personalisierungskonzepte anpassbar.
9. In einer prototypischen Implementierung konnten entwickelte Konzepte für ein personalisiertes universelles Content Management audiovisueller Daten am Beispielszenario der personalisierte Suche auf Nachrichtenbeiträgen umgesetzt werden.



# Selbständigkeitserklärung

Hiermit versichere ich, dass ich die vorliegende Arbeit selbständig und nur unter Benutzung der angegebenen Quellen angefertigt habe.

Michael Drews

Rostock, 12.05.2002