

Otto-von-Guericke-Universität Magdeburg



Fakultät für Informatik
Institut für Technische und Betriebliche Informationssysteme

Diplomarbeit

Vergleichende Analyse von Anfragesprachen in Multimedia-Datenbanken

Verfasser:

Marco Paskamp

31. August 1999

Betreuer:

Prof. Dr. Gunter Saake, Universität Magdeburg, ITI

Dr. Ingo Schmitt, Universität Magdeburg, ITI

Universität Magdeburg
Fakultät für Informatik
Postfach 4120, D-39016 Magdeburg
Germany

Paskamp, Marco:

*Vergleichende Analyse von Anfragesprachen
in Multimedia-Datenbanken*

Diplomarbeit, Otto-von-Guericke-Universität
Magdeburg, 1999.

Vorwort

Anfragesprachen sind ein wesentliches Merkmal von Datenbank-Management-Systemen und mitverantwortlich für deren Erfolg. Durch den Einzug der Datenbanktechnologie in das Gebiet der Multimedia-Informationssysteme spielt die Verfügbarkeit von speziellen Anfragesprachen für Multimedia-Anwendungen eine ebenso wichtige Rolle.

Eine umfangreiche Aufarbeitung von Ansätzen für Multimedia-Anfragesprachen und deren Vergleich waren das Ziel dieser Arbeit. Dazu mußten zunächst die Besonderheiten von Anfragen an Multimedia-Datenbanksysteme untersucht werden, um daraus geeignete Vergleichskriterien für die verschiedenen Ansätze abzuleiten. Aus der Untersuchung der einzelnen Sprachansätze und den darin enthaltenen Defiziten wurden Designprinzipien als Grundlage für zukünftige Entwicklungen abgeleitet.

Die vorliegende Arbeit entstand im Rahmen meiner Diplomarbeit am Institut für Technische und Betriebliche Informationssysteme der Fakultät für Informatik an der Otto-von-Guericke-Universität Magdeburg im Sommersemester 1999.

Danksagung

An dieser Stelle möchte ich mich besonders bei Dr. Ingo Schmitt für die ausgezeichnete Betreuung meiner Arbeit bedanken. In zahlreichen Konsultationen und durch seine konstruktive Kritik beim Korrekturlesen der Arbeit gab er mir wertvolle Hinweise. Auch Dipl.-Inf. Stephan Dassow danke ich für seine Anmerkungen bei verschiedenen Diskussionen. Weiterhin möchte ich mich bei Prof. Saake für die Unterstützung dieser Arbeit sowie bei Dr. Dobrowolny für die kurzfristige Übernahme des zweiten Gutachtens bedanken. Den Mitarbeitern des Institutes für Technische und Betriebliche Informationssysteme danke ich für ihre stets freundliche Unterstützung bei fachlichen Fragen.

Inhaltsverzeichnis

Abbildungsverzeichnis	vii
Tabellenverzeichnis	ix
Verzeichnis der Abkürzungen	xi
1 Einleitung	1
1.1 Motivation	1
1.2 Aufbau der Arbeit	2
2 Grundlagen	5
2.1 Multimedia-Daten	5
2.2 Metadaten	9
2.3 Multimedia-Datenbank-Management-System	10
2.4 SQL und OQL	13
3 Anfragen an Multimedia-Datenbanken	19
3.1 Inhaltsbasierte Anfragen	20
3.1.1 Datenmodelle für eine Inhaltsbeschreibung	20
3.1.2 Abschwächung/Verschärfung von Anfrageausdrücken	21
3.2 Räumliche Anfragen	23
3.2.1 Räumliche Repräsentation	23
3.2.2 Räumliche Beziehungen	24
3.2.3 Räumliche Operationen	29

3.3	Zeitbezogene Anfragen	30
3.3.1	Zeitliche Beziehungen	30
3.3.2	Zeitbezogene Operationen	32
3.4	Unscharfe Anfragen	33
3.4.1	Der Anfrageprozeß	34
3.4.2	Unschärfe und Gewichtung in Anfrageausdrücken	35
3.5	Präsentation von Anfrageergebnissen	36
3.5.1	Merkmale einer Multimedia-Präsentation	36
3.5.2	Datenmodelle für die Präsentation	37
4	Anforderungen an Multimedia-Anfragesprachen	39
4.1	Allgemeine Anforderungen an Anfragesprachen	39
4.2	Spezielle Anforderungen an Multimedia-Anfragesprachen	46
4.3	Zusammenfassung	50
5	Ausgewählte Ansätze für Multimedia-Anfragesprachen	53
5.1	Neu entwickelte Anfragesprachen	53
5.1.1	PICQUERY+	54
5.1.2	SCORE	56
5.1.3	TVQL	59
5.1.4	Framework für eine strukturierte Multimedia-Datenbank	63
5.2	Erweiterungen von SQL	68
5.2.1	KEQL	69
5.2.2	CSQL	72
5.2.3	Medienspezifische abstrakte Datentypen und SQL/MM	74
5.2.4	VideoSQL	79
5.3	Erweiterungen von OQL	81
5.3.1	MMSQL	82
5.3.2	MOQL und VisualMOQL	85
5.4	Weitere Ansätze	90

6	Vergleich und Bewertung	93
6.1	Vergleich von Multimedia-Anfragesprachen	93
6.2	Klassifikation von Multimedia-Anfragesprachen	100
6.3	Bewertung der untersuchten Anfragesprachen	102
6.4	Entwurfsprinzipien für Multimedia-Anfragesprachen	104
7	Zusammenfassung und Ausblick	107
7.1	Ergebnisse	107
7.2	Ausblick	108
	Literaturverzeichnis	111
	Thesen	121

Abbildungsverzeichnis

2.1	Grobe Systemarchitektur eines MMDBMS	12
3.1	Beispiel einer Begriffshierarchie	22
3.2	Räumliche Einteilung durch Richtungsbeziehungen	25
3.3	Topologische Beziehungen	28
5.1	Beispiel für die Inhaltsbeschreibung von Bildern in SCORE	58
5.2	Objekte zur Beschreibung von Videodaten in TVQL	60
5.3	Beispiel einer visuellen zeitbezogenen Anfrage in TVQL	62
5.4	Drei-Ebenen des Metadatenmodells von KEQL	70
5.5	Überblick über das DISIMA-Datenmodell	86
6.1	Klassifikation von Multimedia-Anfragesprachen	100
6.2	Klassifikation nach Arten von Anfragen	101

Tabellenverzeichnis

3.1	Anfragearten in MMDBMS	19
3.2	Beziehungen im eindimensionalen Raum	26
3.3	Definition der Richtungsbeziehungen	27
3.4	Räumliche Operationen	30
3.5	Zeitbezogene Operationen	32
4.1	Allgemeine Anforderungen für Anfragesprachen	51
4.2	Spezielle Anforderungen für Multimedia-Anfragesprachen	52
5.1	Beispielanfrage in PICQUERY+	56
5.2	Anfrageprädikate in CSQL	74
6.1	Vergleich SQL/OQL-basierter Sprachansätze anhand allgemeiner Kriterien	94
6.2	Vergleich neu entwickelter Sprachansätze anhand allgemeiner Kriterien .	96
6.3	Gegenüberstellung anhand multimediaspezifischer Kriterien	98
6.4	Stärken und Schwächen universeller Multimedia-Anfragesprachen	103

Verzeichnis der Abkürzungen

ADT	Abstrakter Datentyp
ANSI	American National Standards Institute
DBMS	Datenbank-Management-System
ER	Entity-Relationship
IR	Information-Retrieval
ISO	International Standardization Organization
MBR	Minimales umschreibendes Rechteck (<i>englisch</i> : Minimum Bounding Rectangle)
MADT	Medienspezifischer abstrakter Datentyp
MMDB	Multimedia-Datenbank
MMDBMS	Multimedia-Datenbank-Management-System
MMQL	Multimedia-Anfragesprache (<i>englisch</i> : Multimedia Query Language)
ODBMS	Objektdatenbank-Management-System
ODMG	Object Database Management Group
OMG	Object Management Group
OQL	Objektanfragesprache (<i>englisch</i> : Object Query Language)
ORDBMS	Objektrelationales Datenbank-Management-System
QBE	Query By Example
QoS	Quality of Service
SFW	Select-From-Where
SQL	Structured Query Language

1 | Einleitung

Die Verfügbarkeit von geeigneten Datenendgeräten und die ständig zunehmende Kapazität der Speichermedien führt zu einer immer stärkeren Integration von Multimedia-Daten in heutige Kommunikations- und Informationssysteme. Dadurch entstanden in den vergangenen Jahren große Ansammlungen von Multimedia-Daten, für deren Verwaltung leistungsfähige Multimedia-Datenbank-Management-Systeme benötigt werden. Das Einsatzgebiet solcher Datenbanksysteme ist sehr vielseitig. So finden Multimedia-Datenbank-Systeme unter anderem in Geographischen Informationssystemen, Digitalen Bibliotheken und zur Verwaltung von Röntgen- und anderen medizinischen Aufnahmen Verwendung. Zusätzlich ergeben sich durch die zunehmende Verbreitung des Internets und die damit verbundene Verfügbarkeit zahlreicher Multimedia-Informationen weitere Anwendungsgebiete.

Als ein wesentlicher Bestandteil von Multimedia-Datenbank-Management-Systemen sollen in dieser Arbeit Multimedia-Anfragesprachen untersucht werden. Dieses erste Kapitel dient dabei der Einführung in die Thematik. Außerdem werden die Ziele und der Aufbau der Arbeit erläutert.

1.1 Motivation

Im allgemeinen arbeiten Benutzer mit einer Datenbank, indem sie Anfragen stellen oder Veränderungen an den Daten vornehmen. Um Anfragen bezüglich den in der Datenbank abgelegten Informationen stellen zu können, bieten die meisten Datenbanksysteme eine spezielle Sprache (Anfragesprache) an. Mit Hilfe dieser Anfragesprache werden die Struktur und die Auswahl von Daten charakterisiert, ohne einen speziellen Algorithmus zur Berechnung der Anfrageergebnisse vorzugeben. Auch in Multimedia-Datenbanken wird für eine effiziente Suche eine leistungsfähige Anfragesprache benötigt.

Im Gegensatz zu Anfragesprachen in traditionellen Datenbanken werden an Multimedia-Anfragesprachen aber weitaus höhere Anforderungen gestellt. So enthalten Multimedia-Daten unter anderem räumliche und zeitliche Merkmale, die durch eine Anfragesprache berücksichtigt werden müssen. Außerdem basieren Anfragen an Multimedia-Datenbanken häufig auf dem Inhalt der Medienobjekte. Dieser wird mit Hilfe spezieller Algorithmen oder durch den Benutzer aus den rohen Mediendaten abgeleitet. Eine hundertprozentig exakte Inhaltsbeschreibung ist dabei in den meisten Fällen nicht möglich.

Multimedia-Anfragesprachen müssen dies berücksichtigen und benötigen deshalb spezielle Konzepte bei der Anfrageformulierung. Ein weiterer Grund, warum Multimedia-Anfragesprachen benötigt werden, sind die besonderen Anforderungen bei der Darstellung der Anfrageergebnisse. Denn anders als alphanumerische Daten können Multimedia-Daten (z.B. Videosequenzen) nicht in Form von Tabellen angezeigt werden. Die Anfragesprache muß somit zusätzlich Eigenschaften der Präsentation von Anfrageergebnissen festlegen.

Die Notwendigkeit spezieller Anfragesprachen für Multimedia-Datenbanken führte in der Vergangenheit zur Entwicklung einer Reihe von unterschiedlichen Sprachansätzen. Diese, fast ausschließlich im Rahmen verschiedener Forschungsprojekte entstandenen Ansätze, sollen in der vorliegenden Arbeit aufgearbeitet und analysiert werden. Durch einen Vergleich der einzelnen Multimedia-Anfragesprachen soll festgestellt werden, inwieweit diese für eine effektive Suche in Multimedia-Datenbanken geeignet sind. Ein weiteres Ziel der Arbeit stellt die Ableitung von allgemeinen Entwurfsprinzipien für zukünftige Entwicklungen von Multimedia-Anfragesprachen dar. Die Ergebnisse dieser Arbeit sollen als Grundlage für weitere Forschungsaktivitäten dienen.

1.2 Aufbau der Arbeit

Nachfolgend soll kurz der Aufbau der Arbeit vorgestellt werden. Dabei wurde bereits in diesem einleitenden Kapitel die Motivation für die Untersuchung von Anfragesprachen in Multimedia-Datenbanken erläutert. Außerdem wurden die Ziele dieser Arbeit vorgestellt.

Kapitel 2 beschreibt grundlegende Merkmale von Multimedia-Datenbank-Management-Systemen. Des weiteren werden die wesentlichen Eigenschaften von Multimedia-Daten und eine Beschreibung dieser durch geeignete Metadaten erläutert. Da die Standardanfragesprachen SQL und OQL die Ausgangsbasis für viele der im weiteren Verlauf der Arbeit beschriebenen Multimedia-Anfragesprachen sind, wird in Kapitel 2 eine kurze Einführung zu diesen Sprachen gegeben.

Die verschiedenen Arten von Anfragen, die typisch für Multimedia-Datenbanken sind, werden in Kapitel 3 vorgestellt. Weiterhin erfolgt eine Beschreibung grundlegender Konzepte, um die verschiedenen Formen von Anfragen zu unterstützen.

Kapitel 4 stellt als Ausgangsbasis für den späteren Vergleich der einzelnen Ansätze für Multimedia-Anfragesprachen Kriterien auf und diskutiert diese. Dabei werden sowohl allgemeine Anforderungen, welche für jede Art von Anfragesprache gelten, als auch multimediaspezifische Anforderungen untersucht.

Einen umfassenden Überblick über die verschiedenen Forschungsansätze für Multimedia-Datenbanken gibt das Kapitel 5 dieser Arbeit. Dabei wird neben den wichtigsten Konzepten der einzelnen Sprachansätze auch das zugrunde liegende Datenmodell betrachtet, da dieses für die Semantik der Anfragekonstrukte eine wesentliche Rolle spielt.

Unter Verwendung der aufgestellten Kriterien erfolgt in Kapitel 6 ein Vergleich der Multimedia-Anfragesprachen und eine Bewertung ihrer Leistungsfähigkeit. Aufbauend

auf diesem Vergleich werden die Sprachansätze klassifiziert. Außerdem wird untersucht, inwieweit die Anfragesprachen für die Verwendung in Multimedia-Datenbanken geeignet sind. Eine Auflistung von Entwurfsprinzipien für zukünftige Entwicklungen von Multimedia-Anfragesprachen schließt das Kapitel ab.

Eine Zusammenfassung der Ergebnisse dieser Arbeit und ein kurzer Ausblick auf mögliche weitere Forschungsarbeit im Zusammenhang mit Multimedia-Anfragesprachen in Kapitel 7 beschließen diese Arbeit.

2 | Grundlagen

Als Ausgangsbasis für die Untersuchung von Multimedia-Anfragesprachen wird in diesem Kapitel das Umfeld der Arbeit untersucht. Zwei Gebiete werden dabei besonders betrachtet: die Multimedia-Daten und deren Eigenschaften sowie die grundlegenden Merkmale eines Multimedia-Datenbank-Management-Systems. Des Weiteren wird die Beschreibung des Inhaltes von Multimedia-Daten durch geeignete Metadaten vorgestellt. Am Ende des Kapitels erfolgt eine kurze Einführung in die Standardanfragesprachen SQL und OQL, da diese Ausgangsbasis für viele der im weiteren Verlauf der Arbeit vorgestellten Multimedia-Anfragesprachen sind.

2.1 Multimedia-Daten

Dieser Abschnitt definiert grundlegende Begriffe, die im Zusammenhang mit Multimedia-Daten verwendet werden, und beschreibt wesentliche Eigenschaften von Multimedia-Daten. Dabei stellt sich am Anfang einer solchen Betrachtung die Frage: Was ist ein Medium? Nach [MW91] versteht man unter einem *Medium* einen Nachrichtenträger, welcher über einen Vorrat an Darstellungsmitteln verfügt, mit denen eine Nachricht gestaltet oder formuliert werden kann.

Eine Klassifikation von Medien kann nach den menschlichen Sinnesorganen (sehen, hören, tasten, schmecken, riechen) sowie nach sprachlichen und nicht-sprachlichen Medien erfolgen. Häufig werden Medien auch nach ihrer Abhängigkeit von der Zeit in *continuous Medien* (zeitabhängige Medien) und *non-continuous Medien* (zeitunabhängige Medien) unterschieden.

Populäre Medien im Umgang mit Rechnern sind vor allem Text, Grafik, Bild, Audio und Video. In diesem Zusammenhang wird häufig auch der Begriff *Medientyp* für eine Klasse bestimmter Medien und *Medienobjekt* für ein Datenobjekt, das eine Nachricht in einem bestimmten Medium enthält (z. B. ein Text), verwandt. Bei der Zusammenstellung mehrerer verschiedener Medien spricht man von *Multimedia*. Analog dazu ist ein *Multimediaobjekt* ein Datenobjekt, das aus mehreren Medienobjekten zusammengesetzt ist. Nachfolgend sind die in Multimedia-Datenbanken am häufigsten vorkommenden Medientypen und deren charakteristische Merkmale aufgeführt.

Text

In fast allen Anwendungen werden Texte für die Darstellung und Verarbeitung von Informationen benutzt und stellen somit einen sehr wichtigen Medientyp dar. Ein Text besteht dabei aus einer endlichen Folge von druckbaren Zeichen und speziellen Steuerzeichen. Im allgemeinen sind Texte in Dateien abgelegt, wobei das Format dieser Dateien vom verwendeten Textverarbeitungssystem abhängt. Um Texte in Multimedia-Datenbanken abzulegen, müssen diese unter Umständen von verschiedenen Dateiformaten in eine interne Darstellung konvertiert werden. Häufig beschränkt sich diese interne Darstellung auf die Abbildung der Texte durch Zeichenketten. Texte besitzen aber oft auch eine Struktur (Kapitel, Abschnitt, Unterabschnitt usw.) und enthalten meist weitere Informationen wie Autor, Titel, Zusammenfassung und Schriftart. Auch diese Informationen sollten zur Unterstützung einer leistungsfähigen Suche aus den Textdateien abgeleitet und im Datenbanksystem abgelegt werden. Für die Beschreibung der Struktur eines Textes existieren spezielle Sprachen, wie zum Beispiel SGML [ISO86].

Gesucht wird in Texten im allgemeinen nach dem Vorkommen bestimmter Wörter und Wortgruppen. Die Suche kann dabei auch auf bestimmte Textteile, zum Beispiel die Überschriften, eingeschränkt werden. Probleme treten in diesem Zusammenhang durch Synonyme¹ und Homonyme² auf. Deshalb werden zusätzlich zur Volltextsuche Schlagwortlisten und Thesauren verwendet.

Grafik

In Anlehnung an [MW91] wird unter einer Grafik eine Zeichnung (Vektorbild) verstanden, welche aus einer komplexen Struktur von geometrischen Objekten, Beschriftungen, Schattierungen und Texturen besteht. Typische Beispiele für Grafiken sind technische Zeichnungen, Flußdiagramme, Geschäftsgrafiken, Landkarten oder mathematische Kurven. Allgemein kann eine Grafik als eine ungeordnete Menge von Linien aufgefaßt werden, welche sich durch zwei- oder dreidimensionale Koordinaten zu geometrischen Objekten höherer Ordnung (Polygone bzw. Polyeder) zusammenfügen. Den Linien können dabei Attribute zugeordnet sein, wie Linienart, Linienstärke und Farbe. Um eine Grafik in einer Datenbank abzulegen, muß deren Struktur aus verschiedenen Dateiformaten in eine interne Darstellung abgebildet werden. Anfragen an Grafiken beziehen sich vor allem auf die räumliche Anordnung der einzelnen geometrischen Objekte, den Abstand dieser zueinander und die räumliche Größe von Objekten.

Rasterbild

Rasterbilder repräsentieren digitalisierte Zeichnungen, Bilder und Fotografien. Sie werden intern dargestellt als eine Matrix von Bildpunkten, welche bei der Ausgabe einen

¹verschiedene Worte haben die gleiche Bedeutung

²das gleiche Wort hat in verschiedenen Zusammenhängen unterschiedliche Bedeutungen

geschlossenen optischen Eindruck erwecken. Die einzelnen Komponenten dieser Matrix werden auch Pixel genannt. Jeder dieser Pixel enthält eine Farbinformation. Entsprechend des Speicherplatzes, der für die Farbinformation eines Pixel reserviert wird, ergeben sich verschiedene Farbtiefen (8, 16 oder 24 Bit). Von dieser Farbtiefe und der Größe der Matrix hängt die Qualität des Bildes, aber auch der benötigte Speicherplatz ab. Um den Speicherbedarf möglichst gering zu halten, werden verschiedene Kompressionsverfahren bei der Abspeicherung von Rasterbildern in Dateien angewandt. Die in einer solchen Datei vorliegenden Bilddaten werden nachfolgend auch als rohe Bilddaten (oder Rohdaten) bezeichnet.

Rasterbilder werden in Multimedia-Datenbanken meist als Rohdaten abgelegt. Das dafür verwendete Format muß so gewählt werden, daß beim Im- und Export der Bilder in bzw. aus der Datenbank möglichst kein Informationsverlust auftritt.

Um Anfragen an den Inhalt der Bilder zu ermöglichen, müssen beim Einfügen dieser in die Multimedia-Datenbank aus den Rohdaten zahlreiche Informationen (z. B. Farbverteilung, Gestalt, Größe, räumliche Merkmale, Semantik usw.) automatisch oder durch den Benutzer abgeleitet und in der Datenbank in Form von Attributen und Indexen abgelegt werden. Hierbei werden häufig interessante Bildausschnitte, nachfolgend Bildobjekte genannt, aus dem gesamten Bild ausgewählt und Beziehungen zwischen Bildobjekten bestimmt. Interessante Bildobjekte könnten zum Beispiel zwei Personen *A* und *B* auf einem Foto sein. Die relative räumliche Anordnung dieser Personen (z. B. Person *A* links von Person *B*) ist dabei eine mögliche Beziehung. Weitere wichtige Merkmale, die sich aus den rohen Bilddaten ableiten lassen, werden im Zusammenhang mit der Betrachtung von Metadaten in einem der nachfolgenden Abschnitte noch genauer vorgestellt.

Eine Suche in Rasterbildern erfolgt im allgemeinen anhand räumlicher Merkmale, der Gestalt, der Textur, der Farbverteilung und weiterer inhaltsbeschreibender Merkmale. Häufig wird eine Ähnlichkeitssuche zwischen Rasterbildern durchgeführt. Hierfür werden spezielle Ähnlichkeitsmaße und Berechnungsalgorithmen benötigt, welche aber in dieser Arbeit nicht betrachtet werden. Vorgestellt werden solche Verfahren unter anderem in [Fal99].

Audio

Die Sprache ist das mit Abstand wichtigste Medium zum Nachrichtenaustausch zwischen Menschen. Auch andere akustische Signale, wie Musik und Geräusche, haben eine außerordentliche Bedeutung. Durch die Verfügbarkeit geeigneter Technik kommen akustische Medien auch immer häufiger auf Rechnern zum Einsatz. Audio ist dabei ein Oberbegriff für die verschiedenen akustischen Medientypen.

Im Gegensatz zu den bisher betrachteten Medientypen haben Audiodaten eine zeitliche Charakteristik. Für die Darstellung von Audiodaten im Rechner werden diese in einem festen zeitlichen Abstand digitalisiert. Dieser zeitliche Abstand wird auch Samplingfrequenz genannt. Je höher diese Samplingfrequenz ist, um so besser ist die Qualität des digitalisierten Audiosignals, aber um so höher ist auch der Speicherplatzbedarf. Ähn-

lich wie bei Rasterbildern werden bei der Abspeicherung von Audiodaten verschiedene Komprimierungsverfahren eingesetzt, um das hohe Datenvolumen zu minimieren. Die dabei entstehenden Audiodateien werden als Audio-Rohdaten bezeichnet.

In Multimedia-Datenbanken werden zur Verwaltung von Audiodaten meist die Rohdaten abgespeichert, aus denen eine Reihe von Merkmalen abgeleitet werden. Dazu zählen unter anderem die Lautstärke, die Tonlage (Pitch) und der Klang eines Audiosignals. Außerdem wird der Inhalt durch Metadaten beschrieben. Ein Audiosignal kann in einzelne zeitliche Abschnitte zerlegt werden. Zwischen diesen Abschnitten können verschiedene zeitliche Beziehungen, z.B. zeitliche Überlappung, auftreten. Eine Suche in Audiodaten erfolgt meist über die Metainformationen und die zeitlichen Beziehungen. Teilweise wird aber auch eine Ähnlichkeitssuche anhand bestimmter akustischer Merkmale angeboten.

Eine spezielle Form von Audiodaten ist die Sprache. Diese gewinnt eine immer höhere Bedeutung in Multimedia-Datenbanken. Es existieren verschiedene Verfahren, um den Inhalt von Sprachdokumenten zu indizieren, und damit eine Suche nach bestimmten Worten und Wortgruppen, ähnlich der Freitextsuche in Textdokumenten, zu ermöglichen. Ein solches Verfahren wird unter anderem in [Sch97] vorgestellt.

Video

Videodaten sind eine beliebig lange, endliche Folge von Rasterbildern oder Grafiken mit einer zugeordneten Tonspur. Ein Video kann als Aggregation von Bild und Audio betrachtet werden und ist somit ebenfalls zeitabhängig. Dabei besteht zwischen den einzelnen Bildern eines Videos eine strenge Ordnung und zu den Audiodaten der Tonspur ein straffer zeitlicher Bezug (Synchronisation). Während das Datenvolumen bei Bild- und Audiodaten bereits sehr hoch war, steigt es durch die Kombination dieser beiden in Videodaten nochmals deutlich an. Um Videodaten in Rechnern abspeichern zu können, werden deshalb effektive Datenkompressionsverfahren benötigt.

Für die Verwaltung von Videodaten in Multimedia-Datenbanken erfolgt ähnlich wie für Bild- und Audiodaten eine Abspeicherung der rohen Videodaten. Anschließend werden daraus geeignete Metainformationen abgeleitet. Neben den bereits von Bild- und Audiodaten bekannten Merkmalen wird vor allem versucht, Szenen³ und Kamerabewegungen in Videos zu erkennen. Ein Video kann dadurch in einzelne Segmente zerlegt werden. Zwischen den Segmenten bestehen zeitliche Abhängigkeiten. Vier Arten von Anfragen sind typisch für Videodaten:

- Segmentanfragen: Suche nach Szenen zu einer Bedingung, zum Beispiel: „Finde alle Stellen im Video, wo John Wayne auf dem Pferd sitzt.“,
- Objektanfragen: Suche nach Objekten in einem Video oder einem Videosegment, zum Beispiel: „Zeige alle Personen die im Video vorkommen.“,

³Videoausschnitt, welcher eine bestimmte abgeschlossene Handlung enthält

- **Aktivitätsanfragen:** Suche nach Handlungen in einem Video oder einem Videosegment und
- **Eigenschaftsanfragen:** Suche nach Video oder Videosegment anhand spezieller Eigenschaften, z. B. zeitliche Beziehungen zwischen einzelnen Szenen.

2.2 Metadaten

Wie bei der Beschreibung der verschiedenen Multimedia-Daten bereits mehrfach erläutert, besitzen diese ein sehr großes Datenvolumen und werden meist als rohe Mediendaten abgelegt. Inhaltsbasierte Anfragen auf Basis der rohen Mediendaten sind aber nur sehr schwer oder teilweise gar nicht möglich. Deshalb erfolgt häufig eine zusätzliche Beschreibung der Multimedia-Daten durch Metadaten. Nach [Pra97a] sind Metadaten aus den Medienobjekten abgeleitete Daten, welche den Inhalt, die Struktur und die Semantik dieser beschreiben.

Multimedia-Daten in Datenbanken bestehen somit nicht nur aus den Rohdaten, sondern bekommen zusätzlich eine Reihe von Metadaten zugeordnet. Diese Metadaten werden aus Effizienzgründen meist beim Einfügen der Multimedia-Daten in die Datenbank abgeleitet, wobei drei Verfahren zur Gewinnung von Metadaten unterschieden werden:

- *Manuelle Ableitung:* Hierbei wird die Beschreibung des Inhaltes der Multimedia-Daten vom Benutzer durchgeführt. Dieser ordnet jedem einzelnen Multimediaobjekt Metadaten zu, die seiner Meinung nach das Objekt inhaltlich gut beschreiben. Häufig stehen hierfür Stichwortlisten und Terminologiebeschreibungen als Hilfsmittel zur Verfügung. Die Qualität der Metadaten ist dabei sehr stark abhängig vom Benutzer, welcher die Metadaten ableitet.
- *Automatische Ableitung:* Der Prozeß der Ableitung von Metadaten wird automatisch durch das System vorgenommen. Dies geschieht unter anderem durch Berechnung von bestimmten charakteristischen Merkmalen aus den Rohdaten der Multimediaobjekte (z. B. Farbverteilung und Gestalt eines Bildes). Solche Merkmale werden nachfolgend auch als Features bezeichnet. Zur Berechnung werden spezielle Algorithmen, meist aus dem Gebiet des Information Retrieval [BR99], benötigt.
- *Halbautomatische Ableitung:* Dies ist eine Kombination der beiden obigen Verfahren. Dabei werden einige Metadaten durch das System bestimmt. Anschließend korrigiert und erweitert der Benutzer diese.

Die Metadaten, welche zur Beschreibung des Inhaltes von Multimedia-Daten benutzt werden, sind sehr vielfältig. Es wurde deshalb mehrfach versucht, Metadaten zu klassifizieren. In der Literatur [BKS98, ATN99] lassen sich dabei verschiedene Einteilungen finden. Mit Hilfe der Klassifikation von Aberer et al. in [ATN99] sollen verschiedene Arten von typischen Metadaten kurz vorgestellt werden. Es wird unterschieden zwischen:

- *Registrierungs- und Identifikationsdaten*: inhaltsunabhängige Daten, die unter anderem das Format und die Kodierung der Multimedia-Daten sowie Informationen über den Ersteller, das Erstellungsdatum und ähnliches enthalten.
- *Präsentationsdaten*: Metadaten zur Beschreibung der Darstellung eines Medienobjektes. Dazu gehören zum Beispiel die „Quality of Service“-Parameter, welche die gewünschte Qualität der Mediendarstellung beschreiben (vgl. Abschnitt 3.5). Präsentationsdaten sind, wie die Registrierungs- und Identifikationsdaten, nur vom Medientyp und nicht Medieninhalt abhängig.
- *Features*: inhaltsabhängige Information, welche mit Hilfe von speziellen Algorithmen aus den Multimedia-Daten berechnet werden, beispielsweise die Farbverteilung und Gestalt von Bildern.
- *Interpretationsdaten*: Dies sind Metadaten, welche direkt aus den Features unter Benutzung einer Wissensbasis abgeleitet werden können. Ein Beispiel hierfür ist die Erkennung von Gesichtsmerkmalen. Aufbauend auf den verschiedenen Bilderkennungs-Features werden die abgeleiteten Featurewerte mit Hilfe einer Wissensbasis als Gesichtsmerkmale interpretiert.
- *Annotationen*: Der Inhalt der Multimedia-Daten wird durch textbasierte Anmerkungen beschrieben. Diese Art von Metadaten wird häufig zur Beschreibung der Handlung in Videoausschnitten verwendet. Annotationen benötigen die Unterstützung durch den Benutzer.
- *Gebietsspezifische Beschreibungen*: Mit Hilfe eines vorgegebenen Vokabulars, Stichwortlisten, Klassifikationen und ähnlichem wird der Inhalt der Multimedia-Daten beschrieben. Auch Begriffshierarchien (Ontologien), welche in Abschnitt 3.1.2 vorgestellt werden, gehören zu dieser Klasse von Metadaten. Genauso wie bei den Annotationen, ist eine Unterstützung durch den Benutzer notwendig.

2.3 Multimedia-Datenbank-Management-System

Eine Anfragesprache ist ein sehr wesentlicher Bestandteil eines Multimedia-Datenbank-Management-Systems (MMDBMS). Für eine ausführliche Betrachtung von Anfragesprachen muß aber erst einmal erläutert werden, was ein MMDBMS ist und welche wesentlichen Anforderungen dieses erfüllen muß.

In einer Vielzahl von Veröffentlichungen [KB95, MW91, AN97, Sub98] wurde in der Vergangenheit versucht, eine Definition für ein Multimedia-Datenbank-Management-System anzugeben. Ganz allgemein läßt sich ein MMDBMS definieren als ein Datenbank-Management-System, das neben formatierten alphanumerischen Daten auch Medienobjekte integriert verwalten kann [MW91]. In [Sub98] ist ein MMDBMS ein Framework für die Verwaltung einer Vielzahl von Medientypen, dargestellt in unterschiedlichen Formaten. Dabei muß ein MMDBMS zusätzlich zu den klassischen Aufgaben eines DBMS, wie

Integration, Transaktionen, Synchronisation, Datensicherung usw., folgende Aufgaben erfüllen:

- *Integration, Komposition und Präsentation von Multimedia-Datentypen:* Neue Multimedia-Datentypen müssen sich in ein MMDBMS integrieren lassen. Außerdem sollten sich verschiedene Medienobjekte zu Multimediaobjekten zusammensetzen lassen. Eine weitere wichtige Aufgabe des MMDBMS's ist die Präsentation der Daten. Aufgrund der Raum- und Zeitabhängigkeit einiger Medientypen ergeben sich komplexe Anforderungen an die Präsentation, welche das MMDBMS erfüllen muß.
- *Verwaltung sehr großer Mengen von Multimedia-Rohdaten:* Ein MMDBMS muß große Mengen von Multimedia-Daten meist in Form von Rohdaten verwalten. Neben der Datenspeicherung im Hauptspeicher und auf Festplatten müssen noch eine Reihe weiterer Medien, wie zum Beispiel Jukeboxen, Magnetbänder usw., unterstützt werden. Das MMDBMS muß dabei einen effizienten Zugriff auf die Daten sicherstellen.
- *Anfrageunterstützung für Multimedia-Daten:* Anfragen an ein MMDBMS können verschiedene Medientypen, Schlüsselworte, Attribute usw. enthalten, wobei die Anfragen häufig unpräzise spezifiziert sind. Um eine effiziente Suche zu ermöglichen, werden deshalb unter anderem hochdimensionale Indexstrukturen, eine Unterstützung von iterativen Anfragen und ein Relevanzfeedback-Mechanismus benötigt. Die verschiedenen Arten von Anfragen werden im nachfolgenden Kapitel 3 noch ausführlich untersucht.
- *Hohe Performance:* Sehr hohe Anforderungen werden an die Performance eines MMDBMS's in bezug auf Speicherverwaltung, Indexierung, Suche und Anfrageoptimierung gestellt, um beispielsweise die Echtzeit-Präsentation von zeitabhängigen Medien (Audio, Video) zu ermöglichen oder um eine effiziente Suche durchzuführen.
- *Möglichkeiten der Benutzerinteraktion:* Die unterschiedlichen Eigenschaften von Medientypen benötigen ebenso unterschiedliche Methoden für den Datenzugriff und zur Darstellung der verschiedenen Medien. Zum Beispiel müssen zur Benutzerinteraktion bei der Präsentation von Continuous-Datentypen durch das MMDBMS Methoden zum Starten, Stoppen sowie zum Vor- und Rücklauf angeboten werden.

Nachdem die Anforderungen erläutert wurden, soll nachfolgend eine typische Systemarchitektur für ein MMDBMS vorgestellt werden. Eine solche Architektur beschreibt den Aufbau und die wichtigsten Komponenten. Abbildung 2.1 zeigt in Anlehnung an [Pra97a] die grobe schematische Darstellung einer typischen Systemarchitektur für ein MMDBMS. Diese kann unterteilt werden in Multimedia-Datenbank-Server und

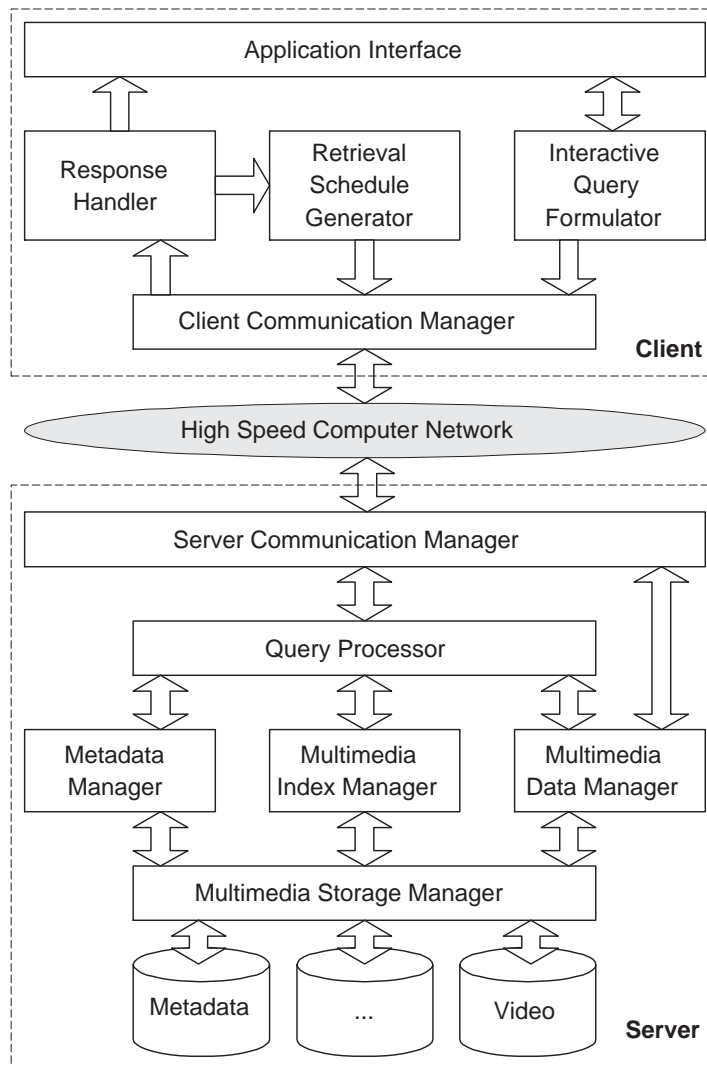


Abbildung 2.1: Grobe Systemarchitektur eines MMDBMS

Multimedia-Datenbank-Client. Die Verbindung zwischen Client und Server erfolgt dabei über ein leistungsfähiges Netzwerk. Entsprechend der Einteilung nach Client und Server sollen nachfolgend die einzelnen Komponenten erläutert werden. Ein typischer Multimedia-Datenbank-Server enthält die folgenden Komponenten:

- *Storage Manager*: verwaltet die Abspeicherung und den Zugriff auf die verschiedenen Medienobjekte. Außerdem werden die Index- und Metadaten mit den entsprechenden Medienobjekten verknüpft.
- *Metadata Manager*: organisiert die Generierung und Aktualisierung der Metadaten, welche den Medienobjekten zugeordnet sind. Weiterhin stellt der Metadata Manager relevante Informationen für den Query Processor zur Anfragebearbeitung bereit.

- *Index Manager*: verwaltet geeignete Indexstrukturen zur inhaltsbasierten Suche und für den effizienten Datenzugriff.
- *Multimedia Data Manager*: unterstützt das Einfügen und die Modifikation der Multimedia-Daten. Des weiteren stellt der Multimedia Data Manager die Medienobjekte bereit, welche als Anfrageergebnis zurückgeliefert werden sollen.
- *Query Processor*: führt den Anfrageprozeß aus. Dazu wird die Anfrage ausgewertet und eine Optimierung durchgeführt. Der Query Processor nutzt hierfür relevante Informationen, die er vom Metadata-, Index- und Multimedia Data Manager zur Verfügung gestellt bekommt. Die Anfrageergebnisse werden unter Beachtung von Präsentationsanforderungen bereitgestellt.
- *Communication Manager*: organisiert den Datentransfer der Anfragen und Anfrageergebnisse über das Netzwerk. Hierfür reserviert der Communication Manager die benötigte Bandbreite und überwacht die Einhaltung der „Quality of Service“-Parameter.

Der Multimedia-Datenbank-Client besteht aus folgenden Komponenten:

- *Communication Manager*: verwaltet den Datenaustausch auf der Clientseite. Die Funktionen entsprechen denen des Server Communication Managers.
- *Retrieval Schedule Generator*: legt einen Schedule für die Übertragung der Mediendaten vom Server fest und steuert die Präsentation der Multimedia-Daten. Der Retrieval Schedule Generator erhält hierzu vom Response Handler Informationen über die Medienobjekte und die damit verknüpften räumlichen und zeitlichen Beziehungen.
- *Response Handler*: empfängt die Anfrageergebnisse und analysiert diese. Je nach Art des Anfrageergebnisses wird dieses zur Präsentation an den Retrieval Schedule Generator oder für eine Modifikation der Anfrage an den Benutzer weitergeleitet.
- *Interactive Query Formulator*: unterstützt den Benutzer bei der Formulierung von Anfragen durch grafische Methoden, durch eine Anfragesprache oder ähnliches.

Im Rahmen dieser Arbeit kann nur ein kurzer Überblick über den Aufbau und die Funktion eines Multimedia-Datenbank-Management-Systems erfolgen. Eine umfassende Behandlung dieses Themas findet sich in der Literatur unter anderem in [KB95, Sub98, ABH97, Pra97a, SH99].

2.4 SQL und OQL

Im Zusammenhang mit Anfragesprachen für MMDBMS spielen die Standardanfragesprachen SQL und OQL eine besondere Rolle, denn viele der im Rahmen dieser Arbeit

untersuchten Ansätze für Multimedia-Anfragesprachen erweitern eine dieser beiden Sprachen um Konstrukte zur Formulierung von Anfragen an Multimedia-Daten. Die wesentlichen Merkmale von SQL und OQL und die grundlegenden Konstrukte dieser Sprachen sollen deshalb in diesem Abschnitt kurz eingeführt werden. Dabei wird zunächst die Anfragesprache SQL im Zusammenhang mit aktuellen Entwicklungen im Rahmen der SQL3-Norm vorgestellt. Im zweiten Teil dieses Abschnittes erfolgt eine kurze Einführung in die Anfragesprache OQL.

SQL

Die Sprache SQL (Structured Query Language) ist die Norm-Datenbanksprache für relationale Datenbanksysteme. Sie dient sowohl zur Datendefinition, Speicherstruktur-Definition, Datenmanipulation und Sichtdefinition als auch zur Anfrageformulierung. SQL wurde 1986 erstmals vom American National Standards Institute (ANSI) und der internationalen Normierungsorganisation (ISO) standardisiert. Da SQL mittlerweile sehr weit verbreitet und allgemein bekannt ist, wird auf eine ausführliche Vorstellung der Konstrukte dieser Sprache verzichtet und hierfür auf einschlägige Literatur verwiesen [DD97]. Nachfolgend sind die wichtigsten SQL-Klauseln der einzelnen Teilsprachen in Anlehnung an [HS97] zusammengefaßt aufgeführt:

- *Datendefinition:* Mit der **create table**-Anweisung werden Relationenschemata mit den dazugehörigen Attributen und Datentypen definiert.
- *Speicherstruktur-Definition:* Mit Hilfe der Anweisungen **create index** und **drop index** können Zugriffspfade für zuvor definierte Relationen angelegt bzw. gelöscht werden.
- *Anfrage:* Durch den **select-from-where**-Block (nachfolgend auch „SFW“-Block genannt) werden die Anfragen formuliert. Dieser besteht im wesentlichen aus folgenden Bestandteilen:
 - **select** beschreibt das Ergebnisschemata durch die Auswahl von Attributen und arithmetischen Operationen sowie Aggregatfunktionen auf den Attributen,
 - **from** legt die zu verwendenden Relationen und eventuell benutzte Variablen für Relationen fest, die als kartesisches Produkt kombiniert werden,
 - **where** definiert Selektions- und Verbundbedingungen,
 - **group by** gruppiert nach bestimmten Attributwerten und
 - **having** definiert Selektionsbedingungen für Gruppen.

Des Weiteren können mehrere der obigen „SFW“-Blöcke (ab SQL92 beliebig) geschachtelt und die Ergebnisse dieser sortiert und vereinigt werden.

- *Datenmanipulation*: Durch die Konstrukte **insert**, **update**, **delete** können Tupel in Relationen eingefügt, verändert oder gelöscht werden.

Eine Weiterentwicklung von SQL-89 und SQL-92 ist die SQL3-Norm, das aktuelle Normierungsprojekt der ANSI und ISO. Dieser zukünftige Standard für Datenbanksprachen versucht die Vorteile von Objektdatenbanken in den SQL-Standard einfließen zu lassen, ohne bewährte SQL-Konzepte aufgeben zu müssen. SQL3 verfügt über etliche neue Konstrukte gegenüber dem bisherigen Standard. Sehr wichtig im Zusammenhang mit Multimedia-Datenbanken sind die objektorientierten Erweiterungen, wie:

- abstrakte Datentypen (ADT), die auch vom Benutzer definiert werden können,
- Objekt-Identifikatoren für Objekt-ADT's und Tupel-Identifikatoren für Tupel in Tabellen,
- ADT-Hierarchien, ähnlich den Typhierarchien in objektorientierten Programmiersprachen,
- Tabellen-Hierarchien, wobei alle Attributnamen und Schlüssel vererbt, aber auch verändert werden können und
- komplexe Datentypen, wie **list** (Listen) und **multiset**(Multimengen).

Besonders interessant für Anfragesprachen in MMDBMS ist das Konzept der benutzerdefinierten abstrakten Datentypen. Ein solcher ADT erweitert das Datenmodell um zusätzliche Datentypen und Methoden und bietet somit unter anderem die Möglichkeit, spezielle Medientypen in das DBMS zu integrieren. Diese können bei der Anfrageformulierung benutzt werden, wodurch SQL zu einer Multimedia-Anfragesprache erweitert wird. Für die Definition eines ADT's wird die bekannte SQL-Definitionssprache erweitert. Das nachfolgende Beispiel zeigt, angelehnt an [SST97], die Definition des abstrakten Datentyps.

```
create type Point
(
  public x real, y real,
  public function point (p real, q real) returns Point
  begin
    declare temp Point;
    set temp = Point();
    set temp..x = p;
    set temp..y = q;
    return temp;
  end,
  public function distance (p1 Point, p2 Point) returns real
    return square_root(square(p1..x - p2..x) +
      square(p1..y - p2..y))
)
```

In dem Beispiel wird ein Datentyp Punkt und dazugehörig eine Distanzfunktion definiert. Ein *Konstruktor* mit dem Namen des ADT's und *Destruktor* mit dem Standardnamen **destroy** werden implizit festgelegt. Außerdem erhält jedes Attribut eines ADT's automatisch eine *Observer-Funktion* zum Lesen und eine *Mutator-Funktion* zum Ändern des Attributwertes. Zur Überprüfung zweier Instanzen eines ADT's auf Gleichheit ist die Standardfunktion *equals* (in SQL durch das Symbol '=' dargestellt) bereits vordefiniert, welche aber auch überschrieben werden kann. Eine Ordnungsrelation, die unter anderem zum Sortieren benötigt wird, muß im Bedarfsfall explizit definiert werden.

Weitere Konstrukte von SQL3 sollen an dieser Stelle nicht betrachtet werden, da sie für die Untersuchung von Multimedia-Anfragesprachen nur eine untergeordnete Rolle spielen. Für zusätzliche Informationen zur SQL3-Norm sei auf [SST97, DD97, Heu97] verwiesen. Im Zusammenhang mit der Untersuchung von Multimedia-Anfragesprachen in Abschnitt 5.2.3 werden multimediaspezifische Erweiterungen von SQL3 vorgestellt.

ODMG-OQL

Dieser Abschnitt befaßt sich mit der Anfragesprache OQL (Object Query Language), welche im Rahmen des ODMG-93-Standards der OMG (Object Management Group) vorgestellt wurde. OQL baut zum Teil auf dem von SQL bekannten „SFW“-Block auf, wobei dem Entwurf folgende Prinzipien zugrunde liegen:

- Die Sprache basiert auf dem Objektmodell der ODMG [Cat94].
- Im Sinne klassischer Anfragesprachen ist OQL ohne Methodenaufrufe deklarativ, optimierbar und nicht berechnungsvollständig.
- Im Gegensatz zu SQL unterstützt die Anfragesprache OQL auch Objektkollektionen, Objektidentifikatoren, Pfadausdrücke, Kollektionsdatentypen und Methoden in Anfragen.
- Zum Erzeugen, Löschen und Ändern von Datenbankobjekten stehen nicht die Standard-SQL-Konstrukte zur Verfügung. Hierfür müssen entsprechende Methoden explizit programmiert und aufgerufen werden.
- Der Sprachentwurf von OQL ist vollständig orthogonal.

Neben dem bekannten „SFW“-Block bietet OQL eine Reihe weiterer syntaktischer Formen für eine Anfrage an, wie Literale, einfache Ausdrücke, Kollektions- oder Mengenausdrücke usw. So ist bereits der Ausdruck: „p.image.name“, eine gültige Anfrage. Wie dieser Ausdruck zeigt, lassen sich innerhalb einer Anfrage auch Pfadausdrücke verwenden, um innerhalb von Tupeln zu den Tupelkomponenten navigieren und Referenzen verfolgen zu können.

Wichtig im Zusammenhang mit Anfragen in Objektdatenbanken ist die Semantik der Anfrageergebnisse. OQL unterstützt relationale, objekterzeugende und objekterhaltende Anfragen. Ein relationales Anfrageergebnis ist, analog zu SQL-Anfragen, eine

Menge von Tupeln. Objektgenerierende Anfragen konstruieren neue Objekte als Anfrageergebnis, wogegen objekterhaltende Anfragen Objektkollektionen bereits vorhandener Objekte darstellen.

Neben den bereits vorgestellten Konzepten unterstützt OQL auch Verbund-Operationen, Quantoren (**for all**, **exists**), Sortierung (**order by**) und Gruppierung (**group by**) von Anfrageergebnissen, Aggregatfunktionen (**min**, **max**, **count** usw.), Konvertierungsausdrücke (**listtoiset**, **flatten**) und benannte Anfragen (**define as**). Weitere Informationen zu OQL kann man in [Cat94, SST97, YM98] finden.

3 | Anfragen an Multimedia-Datenbanken

Daß Multimedia-Daten im Vergleich zu alphanumerischen Daten, wie sie von traditionellen DBMS bekannt sind, einige zusätzliche Eigenschaften besitzen, wurde bereits bei der Beschreibung der verschiedenen Medientypen in Abschnitt 2.1 festgestellt. Beispielsweise enthalten sie häufig räumliche Beziehungen und sind oft auch zeitabhängig. Der vielzitierte Ausspruch, „Ein Bild sagt mehr als tausend Worte.“, zeigt bereits, daß der Inhalt von Multimedia-Daten durch Text allein nur schwer zu beschreiben ist. Dies gilt natürlich ebenso für die Formulierung von Anfragen an Multimedia-Daten. Eine mächtige Anfragesprache für MMDBMS muß deshalb neben exakten Anfragen auf alphanumerischen Daten eine Reihe weiterer Konzepte zur Anfrageformulierung unterstützen. In Tabelle 3.1 sind die wesentlichen Arten von Anfragen in MMDBMS zusammengefaßt.

Anfragetyp	Erläuterung
Inhaltsbasierte Anfragen	Suche anhand des Inhaltes der Medienobjekte
Räumliche Anfragen	Anfragen an die räumliche Anordnung von Objekten und deren Beziehungen zwischen und innerhalb der Medienobjekte
Zeitbezogene Anfragen	Suche über zeitliche Zusammenhänge innerhalb und zwischen den Medienobjekten
Unschärfe Anfragen	unschärfe Formulierung einer Anfrage unter Ausnutzung von Methoden des Information-Retrieval

Tabelle 3.1: Anfragearten in MMDBMS

Im Zusammenhang mit den verschiedenen Arten von Anfragen muß das jeweils zugrunde liegende Datenmodell berücksichtigt werden. Es definiert Konzepte zur Beschreibung der Struktur der Daten. Folgende Konzepte werden unterschieden [HS97]:

- *statische Eigenschaften*, wie Objekte und Beziehungen zwischen diesen,
- *dynamische Eigenschaften*, wie Operationen und
- *Integritätsbedingungen* für Objekte und Operationen.

Das Datenmodell eines MMDBMS besteht aus verschiedenen Teilmodellen zur Beschreibung der einzelnen Eigenschaften von Multimedia-Daten. So dient zum Beispiel ein

räumliches Modell zur Abbildung der räumlichen Eigenschaften, und die Zeitabhängigkeiten werden durch ein zeitbezogenes Modell dargestellt. Diese Datenmodelle werden im Zusammenhang mit den Anfragearten in den nachfolgenden Abschnitten vorgestellt.

3.1 Inhaltsbasierte Anfragen

Im Gegensatz zu Daten konventioneller DBMS ist der Inhalt von Multimedia-Daten im allgemeinen unstrukturiert und liegt nicht in Form von expliziten Attributen eines Datenbank-Schemata vor. Wie bereits im Zusammenhang mit der Inhaltsbeschreibung durch Metadaten in Abschnitt 2.2 erläutert wurde, existieren verschiedene Techniken, um Informationen über den Inhalt aus den Multimedia-Daten abzuleiten, wie zum Beispiel die Extraktion von Features. Ziel all dieser Techniken ist es, dem Benutzer eines MMDBMS eine detaillierte Suche anhand des Inhaltes der Multimedia-Daten zu ermöglichen. Anfragen, die eine solche Suche unterstützen, werden nachfolgend als inhaltsbasierte Anfragen bezeichnet. In anderen Arbeiten wird auch der Begriff inhaltsorientierte oder inhaltsadressierte Anfrage [MW91] verwendet. Welche Art der Suche mit diesen Bezeichnungen gemeint ist, soll durch einige Beispiele verdeutlicht werden:

- Suchen nach Röntgenaufnahmen, auf denen ein Gehirntumor zu sehen ist,
- Recherche in einem Multimedia-Stadtarchiv nach Texten und Bildern über die Kirchen der Stadt,
- Suche nach Fotos in einer Verbrecherkartei, auf denen Personen mit „Vollbart“ und „langen, lockigen Haaren“ zu sehen sind,
- Recherche in einem Nachrichtenarchiv nach Videoaufnahmen, welche Überschwemmungskatastrophen der letzte 10 Jahre in Deutschland zeigen.

In den nun folgenden zwei Unterabschnitten sind die grundlegenden Ansätze für die Modellierung des Inhaltes von Multimedia-Daten aufgelistet. Des weiteren erfolgt eine Betrachtung der Möglichkeiten für eine Abschwächung bzw. Verschärfung von Anfrageausdrücken, um die inhaltsbasierte Suche zu verbessern. Die in den sich anschließenden Abschnitten 3.2 und 3.3 vorgestellten räumlichen und zeitbezogenen Anfragen sind genau genommen auch inhaltsbasierte Anfragen, denn Raum und Zeit stellen ebenfalls Eigenschaften dar, die den Inhalt der Multimedia-Daten beschreiben. Sie werden dennoch gesondert betrachtet, da fast alle Arten von Multimedia-Daten eines dieser beiden Merkmale besitzen und somit die Unterstützung räumlicher und zeitlicher Beziehungen eine Grundanforderung für jedes MMDBMS darstellt.

3.1.1 Datenmodelle für eine Inhaltsbeschreibung

Eine ganz entscheidende Voraussetzung, um inhaltsbasierte Anfragen zu ermöglichen, ist ein geeignetes Datenmodell, welches von den rohen Mediendaten abstrahiert und

den Inhalt der Multimedia-Daten abbilden kann. Eine Klassifikation der verschiedenen Ansätze zur Abbildung des Inhaltes von Multimedia-Daten kann anhand der dazu genutzten Informationen erfolgen. Amato et al. unterscheiden dabei in [AMS98] folgende drei Kategorien:

- *Schlüsselwort-basierte Ansätze*, hierbei wird der Inhalt der Multimedia-Dokumente mittels vom Benutzer hinzugefügter Kommentare beschrieben, zum Beispiel durch Freitext oder durch Schlüsselworte aus einem vorgegebenen Vokabular.
- *Feature-basierte Ansätze* benutzen eine direkt aus den rohen Mediendaten abgeleitete Menge von Features zur Beschreibung des Inhaltes. Typische Features sind dabei Werte, die allgemeine Informationen beschreiben, wie Farbe, Textur, Gestalt, Geschwindigkeit, Position u.Ä. oder auch spezielle anwendungsabhängige Informationen abbilden, zum Beispiel Gesichtmerkmale oder Wettermerkmale auf Satellitenfotos. Die Ableitung der Feature kann manuell durch spezielle Personen oder automatisch durch das DBMS erfolgen.
- *Konzept-basierte Ansätze* interpretieren den durch Features abgeleiteten Inhalt der Multimedia-Dokumente mit Hilfe von anwendungsabhängigem Wissen und stellen Zusammenhänge zwischen den abgeleiteten Medienobjekten her. Eine solche Interpretation wäre zum Beispiel die Verknüpfung von Fotos, Audiodokumenten und Videosequenzen, welche Bild- und Tonaufnahmen des Bundeskanzlers enthalten. Dadurch können medienübergreifend Anfragen formuliert werden, wie zum Beispiel: „Zeige alle Informationen über den Bundeskanzler.“ Im allgemeinen sind die konzept-basierten Ansätze anwendungsabhängig und benötigen die Unterstützung des Benutzers für den Interpretationsprozeß.

Nicht immer sind konkrete Multimedia-Datenmodelle ausschließlich einer dieser drei Kategorien zuzuordnen, vielmehr unterstützen sie mehrere oder gar alle oben aufgeführten Ansätze, um eine detaillierte Modellierung des Inhaltes der Multimedia-Daten und damit verbunden umfangreiche Möglichkeiten zur inhaltsbasierten Suche anzubieten.

3.1.2 Abschwächung/Verschärfung von Anfrageausdrücken

Für inhaltsbasierte Anfragen in MMDBMS werden neben unscharfen Anfragemethoden, welche in einem der nachfolgenden Abschnitte untersucht werden, vor allem Möglichkeiten für die Abschwächung und Verschärfung von Anfragen benötigt. Sinnvoll ist dies zum Beispiel, wenn eine Anfrage zu speziell formuliert wurde und die Ergebnismenge leer ist. Um eine Abschwächung bzw. Verschärfung von Anfragen zu erreichen, wird im allgemeinen eine der folgenden Methoden angewandt:

- *manuelle Abschwächung/Verschärfung*, hierfür muß ein iterativer Anfrageprozeß unterstützt werden, bei der ein Benutzer relevante Objekte spezifiziert. Dieses Verfahren wird in Abschnitt 3.4 beschrieben.

- *automatische Abschwächung/Verschärfung*, bei leeren oder zu umfangreichen Anfrageergebnissen wird durch das DBMS während des Anfrageprozesses automatisch eine Abschwächung bzw. Verschärfung vorgenommen, was nachfolgend erläutert wird.

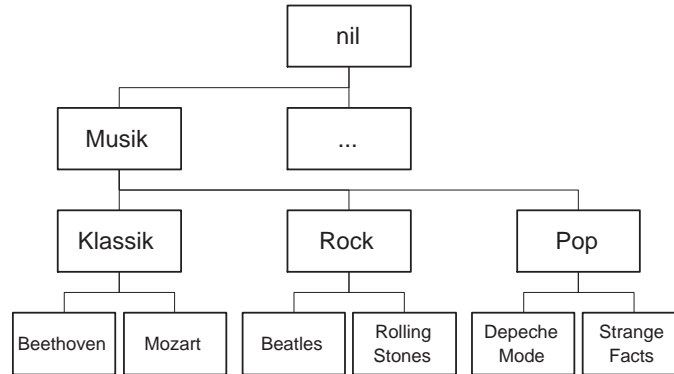


Abbildung 3.1: Beispiel einer Begriffshierarchie

Voraussetzung für die automatische Abschwächung/Verfeinerung von Anfragen ist wiederum eine geeignete Unterstützung durch das Multimedia-Datenmodell. Im einfachsten Fall kann die Darstellung der Beziehungen zwischen den Multimediaobjekten durch eine Begriffshierarchie [OT93] erfolgen. Die Verknüpfungen zwischen den Objekten müssen dabei durch den Anwender vorgegeben werden. Abbildung 3.1 zeigt ein einfaches Beispiel für eine solche Hierarchie. Der allgemeine Begriff „Musik“ unterteilt sich in verschiedene Stilrichtungen (Klassik, Rock und Pop) und weiter in einzelne Musikgruppen. Formal betrachtet kann eine Begriffshierarchie als halbgeordnete Menge (\mathbb{A}, \geq) definiert werden, wobei \mathbb{A} eine Menge von Begriffen ist und \geq eine Halbordnungsrelation. Der Ausdruck $v_1 \geq v_2$ mit $v_1, v_2 \in \mathbb{A}$ sagt dabei aus, daß v_1 allgemeiner als v_2 ist. Bezogen auf das Beispiel in Abbildung 3.1 gelten somit folgende Beziehungen:

- Musik \geq Klassik \geq Beethoven, Musik \geq Klassik \geq Mozart,
- Musik \geq Rock \geq Beatles, Musik \geq Rock \geq Rolling Stones,
- Musik \geq Pop \geq Depeche Mode, Musik \geq Pop \geq Strange Facts.

Mit Hilfe einer solchen Begriffshierarchie können nun die Anfrageausdrücke automatisch verschärft oder abgeschwächt werden, solange die dargestellten Beziehungen dies erlauben und die Begriffe in der Anfrage verwendet werden. So läßt sich im obigen Beispiel bei einer unbefriedigenden Suche nach Objekten mit dem Inhalt „Beatles“ der Anfrageausdruck abschwächen. Gesucht wird nun nach Objekten mit dem Inhalt „Rock“, was eventuell zu einem besseren Suchergebnis führt. Wie schon anhand dieses einfachen Beispiels zu erkennen ist, hängt die Qualität der durch Abschwächung/Verschärfung erreichten Suchergebnisse sehr stark von den abgebildeten Beziehungen in der Begriffshierarchie

ab. Die darin enthaltenen Einträge können neben Attributwerten auch Medienobjekte oder Featurewerte sein.

Das Abschwächen bzw. Verschärfen der Anfrage muß in den Anfrageprozeß integriert werden und ist somit Aufgabe des MMDBMS. Durch den Benutzer muß aber ein Schwellwert vorgegeben werden, bei dessen Erreichen eine Verschärfung des Anfrageausdruckes erfolgen soll. Eine Umsetzung dieses Verfahrens wird unter anderem von V. S. Subrahmanian in [Sub98] vorgestellt. Im Zusammenhang mit dem Entwurf einer Anfragesprache, welche das Abschwächen/Verschärfen von Anfragen unterstützen soll, müssen hierfür geeignete Konstrukte definiert werden.

3.2 Räumliche Anfragen

Viele Anwendungen von MMDBMS basieren auf der Verwendung von räumlichen Beziehungen, welche sowohl zwischen den Medienobjekten als auch innerhalb der Medienobjekte bestehen. Typisch sind dabei Anfragen wie: „Finde alle Bilder, auf denen der Außenminister links vom Bundeskanzler steht.“ Die beiden Personen, Außenminister und Bundeskanzler, stellen dabei zwei Objekte dar, für die eine Beziehung „links von“ durch das räumliche Datenmodell unterstützt werden muß.

Die räumlichen Beziehungen und Operationen des Datenmodells sind die Grundlage für die Formulierung von räumlichen Anfragen und für die Präsentation der Ergebnisse einer Anfrage. Durch die Art und die Anzahl der in den verschiedenen räumlichen Datenmodellen unterstützten Beziehungen werden im wesentlichen die Möglichkeiten bei der Formulierung von räumlichen Anfragen vorgegeben. In der Literatur lassen sich mehrere Modellierungsansätze von räumlichen Beziehungen in MMDBMS finden [HK96, LÖS97, DAPG98]. Eine ausführliche Betrachtung dieser teilweise sehr ähnlichen Ansätze erfolgt in Verbindung mit konkreten Anfragesprachen in Kapitel 5. Ziel dieses Abschnittes ist es, einen allgemeinen Überblick über die räumlichen Beziehungen und Operationen in MMDBMS zu geben. Die nachfolgenden Ausführungen beschränken sich dabei auf eine zweidimensionale Darstellung der räumlichen Beziehungen und Operationen, da diese zur Beschreibung der grundlegenden Zusammenhänge ausreichen. Zum heutigen Zeitpunkt vorgeschlagene Ansätze für MMDBMS verwenden fast ausschließlich eine zweidimensionale Darstellung von räumlichen Zusammenhängen. Für eine dreidimensionale Darstellung bedarf es lediglich einer Erweiterung der nachfolgend vorgestellten Beziehungen und Operationen um eine weitere Dimension.

3.2.1 Räumliche Repräsentation

Ein wichtiger Aspekt, der im Zusammenhang mit räumlichen Beziehungen berücksichtigt werden muß, ist die Darstellung der einzelnen Objekte im räumlichen Datenmodell. Mit Objekt wird in diesem Zusammenhang ein Ausschnitt aus dem gesamten Medienobjekt bezeichnet, wie zum Beispiel eine Person auf einem Bild. Im allgemeinen wird anstelle der Objekte eine vereinfachte Darstellung in der Datenbank abgelegt und zur räumlichen

Indexierung und späteren Suche benutzt. Diese vereinfachte Darstellung wird nachfolgend auch als Region bezeichnet. Abhängig vom Anwendungsgebiet gibt es verschiedene Ansätze für die Beschreibung einer Region. Am häufigsten werden minimale umschreibende Rechtecke verwendet, da sie sich durch nur zwei Punkte definieren lassen und somit einfach zu handhaben sind. Werden aber exakte räumliche Beziehungen benötigt, stößt man bei der Verwendung von umschreibenden Rechtecken auf Schwierigkeiten. So können beispielsweise Überlappungen angezeigt werden, wo gar keine auftreten. Um dieses Problem zu lösen, ist eine präzise räumliche Beschreibung der Objekte nötig. Dies kann durch eine Darstellung der Objekte mittels einer Punktmenge erreicht werden. Dieser Ansatz führt aber zu erhöhten Speicherkosten für die Objekte und zu einem höheren Aufwand für die Berechnung der räumlichen Beziehungen zwischen den Punktmengendarstellungen der Objekte.

Die Auswahl einer geeigneten Beschreibungsmethode muß von Fall zu Fall unterschieden werden. Sie hängt von der durch die Anwendung geforderte Genauigkeit für die räumliche Objektbeschreibung ab, wobei mit zunehmender Genauigkeit der Berechnungsaufwand für die räumlichen Operationen steigt und somit die Effizienz des Gesamtsystems sinkt.

3.2.2 Räumliche Beziehungen

Umfangreiche Untersuchungen von räumlichen Beziehungen zwischen mehreren Objekten erfolgten im Zusammenhang mit Geographischen-Informationssystemen und Bilddatenbanken. Darauf aufbauend existieren mehrere Arbeiten, die versuchen, eine geeignete Menge von räumlichen Beziehungen für MMDBMS zu definieren [DAPG98, LÖS96]. Dabei lassen sich verschiedene Arten von Beziehungen unterscheiden. Diese sind im einzelnen:

- Richtungsbeziehungen,
- topologische Beziehungen und
- Distanzbeziehungen.

Die Richtungsbeziehungen beschreiben die räumliche Lage von Objekten zueinander. In Abbildung 3.2 ist der Zusammenhang zwischen den verschiedenen Richtungsbeziehungen grafisch dargestellt. Die Richtungsangaben beziehen sich dabei auf das Referenzobjekt, welches sich in der Mitte der Abbildung befindet. Die einzelnen Quadrate repräsentieren dabei einen räumlichen Bereich, in dem sich Objekte befinden können, damit die jeweilige Beziehung gilt. Die dargestellten Bereiche sind in bestimmten Richtungen offen (dargestellt durch gestrichelte Linie) und werden nur durch die Systemgrenzen eingeschränkt. Die räumlichen Beziehungen gelten sowohl für eine Darstellung durch minimale umschreibende Rechtecke als auch für Polygon- und Punktmengendarstellungen.

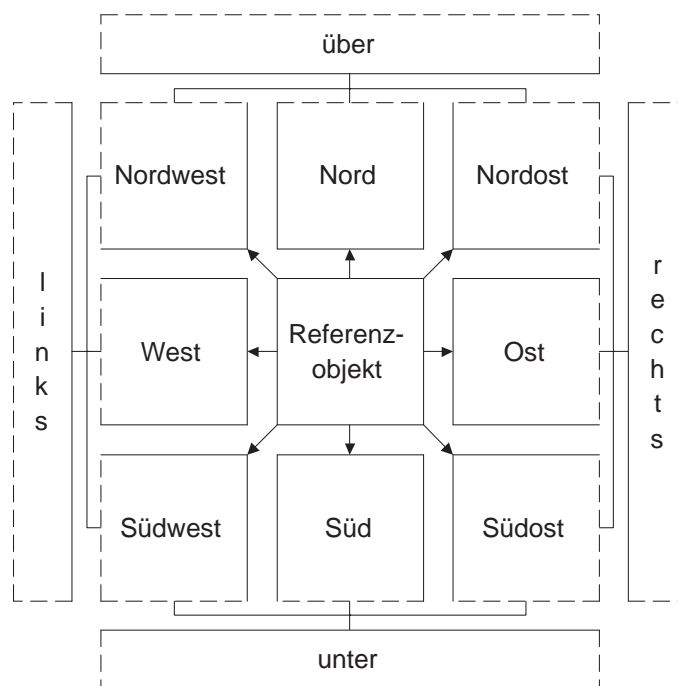


Abbildung 3.2: Räumliche Einteilung durch Richtungsbeziehungen

Prinzipiell kann man auf die allgemeinen Richtungsbeziehungen (links, rechts, über und unter) verzichten. Für eine einfachere Beschreibung, vor allem bei der Anfrageformulierung, sind diese im Ansatz von [LÖS96] zusätzlich enthalten. So wird zum Beispiel für die Beziehung „links“ nur das vertikale Intervall (Nordwest, West, Südwest) betrachtet. Entsprechend kommen für die Beziehungen „über“ und „unter“ nur die horizontalen Intervalle (Nordwest, Nord, Nordost bzw. Südwest, Süd, Südost) in Betracht.

Für eine formale Definition der Richtungsbeziehungen müssen zunächst die Zusammenhänge im eindimensionalen Raum betrachtet werden. Die Objekte sind dabei auf einer Koordinatenachse angeordnet. Der Bereich, den ein Objekt A belegt, kann durch eine Startkoordinate A_s und eine Endkoordinate A_e mit $A_s \leq A_e$ dargestellt werden. Für zwei Objekte A und B sind 13 verschiedene Beziehungen für ihre Anordnung auf der Koordinatenachse möglich. Diese sind in Tabelle 3.2 definiert. Die erste Spalte der Tabelle enthält die Bezeichnung der Relation. In der zweiten Spalte wird für diese Relation ein Symbol definiert, und die dritte Spalte gibt eine formale Definition der Beziehung an.

Aufbauend auf die im eindimensionalen Raum definierten räumlichen Beziehungen können formale Definitionen für die in Abbildung 3.2 dargestellten Richtungsbeziehungen angegeben werden. Tabelle 3.3, welche aus [Li98] entnommen wurde, faßt diese Definitionen zusammen. Für eine bessere Übersichtlichkeit wurde eine verkürzende Schreibweise gewählt. Die geschweiften Klammern entsprechen dabei einer ODER-Verknüpfung der in ihnen angegebenen Relationen. Zum Beispiel ist $A_x\{b, m, o\}B_x$ äquivalent zu




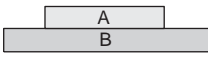

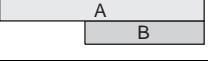



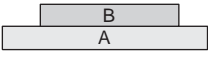
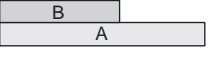
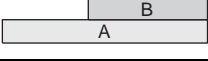
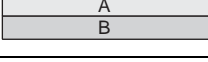
Relation	Symbol	Definition	grafische Darstellung
A before B	b	$A_e < B_s$	
A meets B	m	$A_e = B_s$	
A overlaps B	o	$A_s < B_s < A_e < B_e$	
A during B	d	$B_s < A_s < A_e < B_e$	
A starts B	s	$A_s = B_s \wedge A_e < B_e$	
A finishes B	f	$B_s < A_s \wedge A_e = B_e$	
A before ⁻¹ B	b^{-1}	$A_s > B_e$	
A meets ⁻¹ B	m^{-1}	$A_s = B_e$	
A overlaps ⁻¹ B	o^{-1}	$A_e > B_e > A_s > B_s$	
A during ⁻¹ B	d^{-1}	$A_s < B_s < B_e < A_e$	
A starts ⁻¹ B	s^{-1}	$A_s = B_s \wedge A_e > B_e$	
A finishes ⁻¹ B	f^{-1}	$B_s > A_s \wedge A_e = B_e$	
A equal B	e	$A_s = B_s \wedge A_e = B_e$	

Tabelle 3.2: Beziehungen im eindimensionalen Raum

$A_x b B_x \vee A_x m B_x \vee A_x o B_x$. Durch die Bezeichnung A_x bzw. A_y wird für ein Objekt A die jeweilige räumliche Dimension angegeben, auf die sich eine Relation bezieht.

Während die Richtungsbeziehungen die allgemeine Richtung beschreiben, in welcher sich ein Objekt bezüglich eines anderen befindet, dienen die topologischen Beziehungen der Abbildung von „Nachbarschaftsverhältnissen“ zwischen den Objekten. Eine topologische Beziehung zweier Objekte wäre zum Beispiel eine räumliche Überlappung oder das Enthaltensein eines Objektes in einem anderen. In [LÖS96] werden aufbauend auf einer Arbeit von Pullar und Egenhofer in [EF91, PE88] sechs topologische Beziehungen für MMDBMS vorgeschlagen. Abbildung 3.3 enthält eine grafische Darstellung der nachfolgend formal definierten topologischen Beziehungen:

- A equal B , wenn gilt: $A_x\{e\}B_x \wedge A_y\{e\}B_y$.

Relation	Name	Definition
A südlich von B	Süd	$A_x\{d, d^{-1}, s, s^{-1}, f, f^{-1}, e\}B_x \wedge A_y\{b, m\}B_y$
A nördlich von B	Nord	$A_x\{d, d^{-1}, s, s^{-1}, f, f^{-1}, e\}B_x \wedge A_y\{b^{-1}, m^{-1}\}B_y$
A westlich von B	West	$A_x\{b, m\}B_x \wedge A_y\{d, d^{-1}, s, s^{-1}, f, f^{-1}, e\}B_y$
A östlich von B	Ost	$A_x\{b^{-1}, m^{-1}\}B_x \wedge A_y\{d, d^{-1}, s, s^{-1}, f, f^{-1}, e\}B_y$
A nordwestlich von B	Nordwest	$(A_x\{b, m\}B_x \wedge A_y\{b^{-1}, m^{-1}, o^{-1}\}B_y) \vee$ $(A_x\{o\}B_x \wedge A_y\{b^{-1}, m^{-1}\}B_y)$
A nordöstlich von B	Nordost	$(A_x\{b^{-1}, m^{-1}\}B_x \wedge A_y\{b^{-1}, m^{-1}, o^{-1}\}B_y) \vee$ $(A_x\{o^{-1}\}B_x \wedge A_y\{b^{-1}, m^{-1}\}B_y)$
A südwestlich von B	Südwest	$(A_x\{b, m\}B_x \wedge A_y\{b, m, o\}B_y) \vee$ $(A_x\{o\}B_x \wedge A_y\{b, m\}B_y)$
A südöstlich von B	Südost	$(A_x\{b^{-1}, m^{-1}\}B_x \wedge A_y\{b, m, o\}B_y) \vee$ $(A_x\{o^{-1}\}B_x \wedge A_y\{b, m\}B_y)$
A links von B	links	$A_x\{b, m\}B_x$
A rechts von B	rechts	$A_x\{b^{-1}, m^{-1}\}B_x$
A unter B	unter	$A_y\{b, m\}B_y$
A über B	über	$A_y\{b^{-1}, m^{-1}\}B_y$

Tabelle 3.3: Definition der Richtungsbeziehungen

- A *inside* B, wenn gilt: $A_x\{d\}B_x \wedge A_y\{d\}B_y$.

- A *cover* B, wenn gilt:

$$(A_x\{d^{-1}\}B_x \wedge A_y\{f^{-1}, s^{-1}, e\}B_y) \vee (A_x\{e\}B_x \wedge A_y\{d^{-1}, f^{-1}, s^{-1}\}B_y) \vee$$

$$(A_x\{f^{-1}, s^{-1}\}B_x \wedge A_y\{d^{-1}, f^{-1}, s^{-1}, e\}B_y).$$

- A *overlap* B, wenn gilt:

$$(A_x\{s, d, f\}B_x \wedge A_y\{o, o^{-1}, d^{-1}, f^{-1}, s^{-1}\}B_y) \vee (A_x\{e, d^{-1}, f^{-1}, s^{-1}\}B_x \wedge$$

$$A_y\{o, o^{-1}, s, d, f\}B_y) \vee (A_x\{o, o^{-1}\}B_x \wedge A_y\{o, o^{-1}, f, f^{-1}, d, d^{-1}, s, s^{-1}, e\}B_y).$$

- A *touch* B, wenn gilt:

$$(A_x\{m, m^{-1}\}B_x \wedge A_y\{d, d^{-1}, s, s^{-1}, f, f^{-1}, o, o^{-1}, m, m^{-1}, e\}B_y) \vee$$

$$(A_x\{d, d^{-1}, s, s^{-1}, f, f^{-1}, o, o^{-1}, m, m^{-1}, e\}B_x \wedge A_y\{m, m^{-1}\}B_y).$$

- A *disjoint* B, wenn gilt: $A_x\{b, b^{-1}\}B_x \vee A_y\{b, b^{-1}\}B_y$.

Als letzte Klasse von Beziehungen sollen die Distanzbeziehungen betrachtet werden. Sie beschreiben den räumlichen Abstand von Objekten. Eine einfache Beschreibung dieses Abstandes kann durch Prädikate, wie zum Beispiel „weit entfernt“ oder „in der Nähe

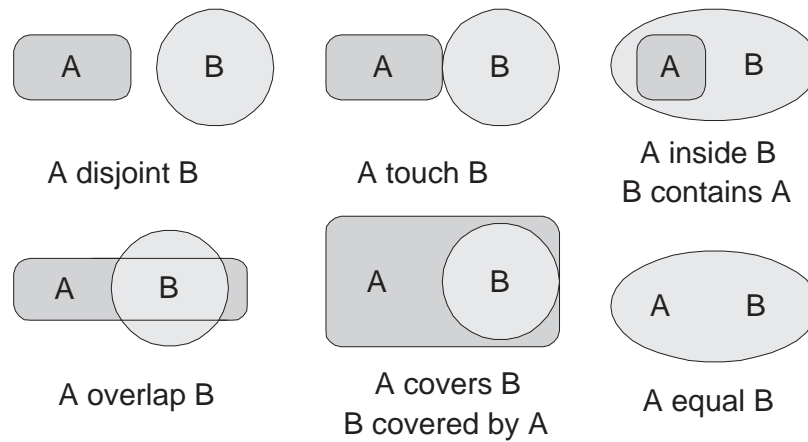


Abbildung 3.3: Topologische Beziehungen

von“, erfolgen. Der Abstand zweier Objekte kann aber auch durch einen konkreten Wert angegeben werden, zum Beispiel „Finde alle Telefonzellen im Umkreis von einem Kilometer.“ Für beide Arten der Beschreibung des Abstandes von Objekten zueinander wird eine Distanzfunktion benötigt. Dies ist eine Funktion L , welche zwei Punkte p und q im Raum R^n auf einen nichtnegativen Wert abbildet. Um diese auf dem Abstand zweier Punkte basierende Distanzfunktion auch für Objekte, welche im allgemeinen durch eine Region (minimales umschreibendes Rechteck oder Punktmenge) repräsentiert werden, verwenden zu können, muß man für die Regionen Referenzpunkte definieren. Eine Möglichkeit wäre, den Mittelpunkt der Region als Referenzpunkt zu verwenden. Die Distanzfunktion muß dabei folgende Bedingungen erfüllen:

1. $L(p, q) = L(q, p)$,
2. $L(p, q) \geq 0$; $L(p, q) = 0$, genau dann wenn $p = q$ und
3. Dreiecksungleichung: $L(p, r) \leq L(p, q) + L(q, r)$.

Die allgemeine Definition der Distanzfunktion D lautet:

$$L(p, q) = \sqrt[m]{\sum_{i=1}^n (|p_i - q_i|)^m}, m \in \mathbb{N}$$

Für verschiedene m sind dabei spezielle Distanzfunktionen definiert. Die bekanntesten und im Zusammenhang mit räumlichen Beziehungen am häufigsten verwendeten Funktionen sind hierbei:

- für $m = 1$ die Manhattan-Distanz L_1 ,
- für $m = 2$ die euklidische Distanz L_2 und
- für $m = \infty$ die maximale Distanz L_∞ .

3.2.3 Räumliche Operationen

Neben der Darstellung von räumlichen Abhängigkeiten enthält ein räumliches Datenmodell für ein MMDBMS zusätzlich Operationen auf den räumlichen Daten. Diese werden benötigt, um zum Beispiel den Anfrageprozeß zu unterstützen, bestimmte Merkmale aus den Mediendaten abzuleiten oder um verschiedene Attribute der räumlichen Objekte zu berechnen. Eine Auflistung möglicher räumlicher Operationen, wie in Tabelle 3.4, kann nicht den Anspruch auf Vollständigkeit erheben, da diese stark vom konkreten Datenmodell der einzelnen MMDBMS abhängen. Zusätzlich benötigen verschiedene Anwendungen spezielle Operationen. Diese zu unterstützen, kann aber im Hinblick auf die Anwendungsunabhängigkeit nicht Aufgabe des MMDBMS sein. Trotzdem sollte eine Grundmenge von allgemeinen, häufig benötigten Operationen als Teil des räumlichen Datenmodells angeboten werden.

Wichtig im Zusammenhang mit Operationen ist die genaue Spezifikation der Typen sowohl für die Operanden als auch für das Ergebnis der Operation. In Tabelle 3.4 wurden die wichtigsten Operationen auf räumlichen Daten und die dazugehörigen Operanden- und Ergebnistypen aufgelistet. Dabei lassen sich anhand des Resultat-Typs drei Arten von Funktionen unterscheiden:

- Testoperationen auf bestimmten räumlichen Beziehungen zwischen den Objekten (vgl. Abschnitt 3.2.2), welche einen Boole'schen Wert berechnen,
- Operationen, die numerische Werte ähnlich den Aggregatfunktionen auf alphanumerischen Daten berechnen und
- verschiedene Mengenoperationen zwischen den Regionen.

Die im unteren Teil der Tabelle 3.4 aufgeführte Durchschnitts-Operation kann als „räumlicher“-Join [SA95], auch Spatial-Join genannt, genutzt werden. Allgemein werden bei einem θ -Join Vergleiche zwischen Attributen durchgeführt, die bezüglich θ vergleichbar sind, wobei θ einen Vergleichsoperator darstellt. Basiert dieser Vergleichsoperator auf räumlichen Beziehungen, so spricht man von einem Spatial-Join. Bei der eingangs erwähnten Durchschnitts-Operation werden zwei Mengen von Regionen bezüglich ihrer räumlichen Lage verglichen, wobei übereinstimmende Paare von Regionen in der Ergebnisregion enthalten sind. Ein Beispiel soll die Bedeutung von Spatial-Joins noch einmal unterstreichen. Gegeben ist eine Menge von Städten und eine Menge von Seen. „Finde alle die Städte, die an einem See liegen.“ wäre eine mögliche Anfrage, welche auf einem Spatial-Join beruht.

Die in Tabelle 3.4 aufgeführten Operationen sind nur eine kleine Auswahl möglicher Operationen. Konkrete Operationen auf räumlichen Daten sind abhängig vom Datenmodell, welches einem konkreten MMDBMS zugrunde liegt.

Name	Operand 1	Operand 2	Ergebnis
equal, touch, inside, overlap, cover, disjoint	Region	Region	Boole'scher Wert
nord, ost, süd, west	Region	Region	Boole'scher Wert
nordost, nordwest, südost, südwest	Region	Region	Boole'scher Wert
links, rechts, über, unter	Region	Region	Boole'scher Wert
Abstand	Region	Region	numerischer Wert
Länge	Linie		numerischer Wert
Umfang	Region		numerischer Wert
Fläche	Region		numerischer Wert
Mittelpunkt	Region		Punkt
Durchschnitt	Region	Region	Region
Vereinigung	Region	Region	Region
Differenz	Region	Region	Region

Tabelle 3.4: Räumliche Operationen

3.3 Zeitbezogene Anfragen

Ein weiteres wesentliches Merkmal von Multimedia-Daten ist deren zeitliche Abhängigkeit. Besonders im Zusammenhang mit den Continuous-Datentypen, wie Audio- und Videosequenzen, ist eine Modellierung der zeitlichen Beziehungen grundlegend für ein MMDBMS. Dies ermöglicht Anfragen wie „Finde alle Stellen in dem Video, wo Person A vor Person B auftaucht.“ Voraussetzung für solche Anfragen ist ein geeignetes zeitliches Datenmodell, in welchem sich die zeitlichen Beziehungen darstellen lassen und welches allgemein benötigte Operationen auf den zeitbezogenen Daten anbietet. Beides soll in den nachfolgenden Abschnitten näher untersucht werden.

Ein wesentlicher Aspekt des zeitlichen Datenmodells ist die Granularität, mit der die Zeit dargestellt wird. Im allgemeinen wird hierbei zwischen diskreten Werten unterschieden, wie Jahr, Monat, Tag, Stunde, Minute, Sekunde und Millisekunde.

3.3.1 Zeitliche Beziehungen

In vielen Multimedia-Daten existieren zeitliche Beziehungen. Diese bestehen sowohl zwischen einzelnen Komponenten innerhalb eines Medienobjektes, zum Beispiel Zeitpunkt des Auftauchens bestimmter Personen in einer Videosequenz, als auch zwischen verschiedenen Medienobjekten. Aufgrund der besonderen Semantik einzelner zeitlicher Abschnitte kann ein zeitabhängiges Medienobjekt in diese Abschnitte zerlegt werden,

welche nachfolgend auch als Sequenzen bezeichnet werden. Ein Beispiel für eine solche Sequenz ist eine Szene in einem Videofilm. Die zeitlichen Informationen, die mit einer Sequenz verknüpft sind, lassen sich dabei durch die nachfolgend aufgelisteten drei Eigenschaften charakterisieren:

- Beginn einer Sequenz,
- Dauer der Sequenz und
- Synchronisation der Sequenz mit anderen Sequenzen.

Angelehnt an diese charakteristischen Eigenschaften von Sequenzen lassen sich drei Zeittypen definieren:

- **Zeitintervall:** Sei $[\Gamma \subset \mathbb{N}, \leq]$ eine Ordnung über die Zeit und a und b zwei Elemente dieser Ordnung mit $a \leq b$. Dann ist die Menge $\{x \in \Gamma | a \leq x \leq b\}$ ein Zeitintervall und a (b) ist die Unter- bzw. Obergrenze des Zeitintervalls.
- **Zeitpunkt:** Ein Zeitpunkt ist ein Sonderfall eines Zeitintervalls, bei dem die Intervalluntergrenze gleich der Intervallobergrenze ist.
- **Zeitspanne:** Der Abstand zwischen der Ober- und der Untergrenze eines Zeitintervalls ist eine Zeitspanne.

Der Zeitpunkt und die Dauer eines Ereignisses können entweder fest oder flexibel beschrieben werden. Fest bedeutet in diesem Zusammenhang, daß die Zeitangaben genau festgelegt sind. Zum Beispiel: „Spiele das Musikstück um 9 Uhr für eine Dauer von 5 Minuten und 23 Sekunden.“ Sind die Zeitangaben flexibel, so kann der Startzeitpunkt und die Dauer unter Einhaltung bestimmter Beziehungen variiert werden, beispielsweise: „Spiele das Musikstück nach dem Ende des Videos, bis das Musikstück beendet ist.“

Eine Beschreibung der zeitlichen Abhängigkeiten zwischen den Multimediaobjekten wird als Synchronisation bezeichnet. Umfangreiche Untersuchungen dazu hat Allen in [All83] durchgeführt und 13 Beziehungen zwischen Zeitintervallen eingeführt. Diese heute allgemein akzeptierten Beziehungen werden als Grundlage für fast alle zeitlichen Modelle in MMDBMS genutzt. Die Beziehungen wurden bereits als räumliche Beziehungen im eindimensionalen Raum in Tabelle 3.2 definiert. Ersetzt man die eindimensionale räumliche Achse durch eine Zeitachse, die systemweit eindeutig ist, so kann diese Definition übernommen werden.

Entsprechend der Darstellung der Zeitintervalle (fest oder flexibel) kann zwischen zeitlich festen Modellen und zeitlich flexiblen Modellen unterschieden werden. Zeitlich feste Modelle werden sehr häufig benutzt, da sie sehr einfach sind. Sie haben aber den Nachteil, daß die zeitlichen Beziehungen für jedes Objekt individuell beschrieben werden, und somit die Beziehungen zwischen den Objekten unberücksichtigt bleiben. Ein typischer Vertreter zeitlich fester Modelle ist das Timeline-Modell [Pra97a, HK96], welches die Multimediaobjekte auf einer Zeitachse anordnet. Zeitlich flexible Modelle

Name	Operand 1	Operand 2	Ergebnis
obere Intervallgrenze	Zeitintervall		Zeitpunkt
untere Intervallgrenze	Zeitintervall		Zeitpunkt
Intervalllänge	Zeitintervall		Zeitspanne
before, meets, starts, finishes, during, overlaps, equal	Zeitintervall	Zeitintervall	Boole'scher Wert
Durchschnitt, Vereinigung, Differenz	Zeitintervall	Zeitintervall	Zeitintervall
Intervall kürzen/strecken	Zeitintervall	Zeitspanne	Zeitintervall
<; >; ≤; ≥; =	Zeitpunkt	Zeitpunkt	Boole'scher Wert
+ oder -	Zeitpunkt	Zeitspanne	Zeitpunkt
Abstand	Zeitpunkt	Zeitpunkt	Zeitspanne
Durchschnitt	Zeitpunkt	Zeitintervall	Zeitpunkt
<; >; ≤; ≥; =	Zeitspanne	Zeitspanne	Boole'scher Wert
+ oder -	Zeitspanne	Zeitspanne	Zeitspanne

Tabelle 3.5: Zeitbezogene Operationen

knüpfen den Startzeitpunkt und die Zeitspanne an bestimmte Bedingungen und an Beziehungen zwischen den Objekten. Zur Modellierung der Beziehungen zwischen den Objekten werden verschiedene Arten von Petri-Netzen [DAPG98] eingesetzt. Dabei werden die Transitionen als Synchronisationspunkte und die Places zur Modellierung von Bedingungen genutzt. Ein weiteres Modell zur Darstellung zeitlich flexibler Beziehungen wird von V. S. Subrahmanian in [Sub98] vorgestellt. Hierbei werden sogenannte „Differenzen Constraints“, eine Menge von einschränkenden Bedingungen der Form: $Startzeitpunkt_Objekt_1 - Endzeitpunkt_Objekt_2 \leq Konstante$, für die Beschreibung zeitlicher Abhängigkeiten zwischen Medienobjekten benutzt. Für eine ausführliche Beschreibung der verschiedenen zeitlichen Modelle sei auf die Literatur verwiesen [DAPG98, Pra97a].

3.3.2 Zeitbezogene Operationen

Zur Unterstützung zeitlicher Anfragen und zur Berechnung bestimmter zeitlicher Eigenschaften der Medienobjekte werden zeitbezogene Operationen benötigt. Für die nachfolgende Auflistung verschiedener Operationen gilt dieselbe Einschränkung wie für die Auflistung räumlicher Operationen in Abschnitt 3.2.3. Es können nur die grundlegenden Operationen betrachtet werden, welche man in jedem MMDBMS benötigt.

In Tabelle 3.5 sind die wichtigsten zeitbezogenen Operationen aufgeführt, wobei eine Einteilung nach den verschiedenen Zeittypen aus dem vorhergehenden Abschnitt vorgenommen wurde. Die Spalten der Tabelle enthalten jeweils den Namen der Operation, die

Typen für die Operanden und den Typ des Ergebnisses der Operation. Bei einstelligen Operationen, wie zum Beispiel die Berechnung der Intervallgrenzen, bleibt das Feld für den zweiten Operanden leer.

Typische Operationen auf Zeitintervallen sind der Test auf bestimmte Beziehungen zu anderen Intervallen (before, meets usw.), Mengenoperationen zwischen mehreren Intervallen und die Berechnung von Intervalleigenschaften, wie Länge eines Intervalls und die Intervallgrenzen. Bei den Mengenoperationen müssen einige Sonderfälle beachtet werden. So ist das Ergebnis der Vereinigung zweier disjunkter Intervalle kein einzelnes Intervall, sondern eine Menge von zwei Intervallen. Ebenso erhält man bei der Berechnung der Differenz zweier Intervalle, wobei das zweite Intervall in dem ersten enthalten ist, als Ergebnis, anstelle eines einzelnen Intervalles, zwei disjunkte Intervalle. Um trotz dieser Abweichungen eine eindeutige Semantik für die Operationen festlegen zu können, behandeln die meisten konkreten Systeme diese Sonderfälle als undefiniert und liefern als Ergebnis einen Nullwert.

Zwei Zeitpunkte bzw. zwei Zeitspannen können jeweils miteinander verglichen werden. Hierzu stehen die Vergleichsoperatoren $<$; $>$; \leq ; \geq ; $=$ zur Verfügung. Außerdem ist eine Addition und Subtraktion einer Zeitspanne zu bzw. von einem Zeitpunkt oder einer Zeitspanne möglich. Für einen Zeitpunkt läßt sich zusätzlich überprüfen, ob er in einem Zeitintervall liegt (Durschnitt), oder der Abstand zwischen zwei Zeitpunkten kann bestimmt werden.

3.4 Unscharfe Anfragen

Anfragen in traditionellen relationalen oder objektorientierten DBMS basieren auf exakten Anfragen, bei denen genau die Tupel bzw. Objekte im Anfrageergebnis enthalten sind, die alle spezifizierten Kriterien des Anfrageausdruckes exakt erfüllen. Genügt ein Datenobjekt nur einem Teil der Kriterien, so ist es nicht in der Ergebnismenge enthalten. Während dieser Ansatz für konventionelle DBMS vollkommen ausreicht, stößt man im Zusammenhang mit Multimedia-Daten auf Probleme, da diese, wie bereits in Abschnitt 2.1 erläutert, sehr komplex sind und größtenteils unstrukturiert vorliegen. Dem Anwender ist es dadurch nicht immer möglich, die gesuchten Multimediaobjekte genau zu spezifizieren. So läßt sich beispielsweise der Inhalt von Bildern, welche gesucht werden, in einer Anfrage nur äußerst schwer oder gar nicht ausdrücken. Des weiteren sind bei exakten Anfragen Objekte, die nur wenig von den Anfrageprädikaten abweichen, schon nicht mehr in der Ergebnismenge enthalten, obwohl sie den Anforderungen des Nutzers genügt hätten. Um diesen Problemen zu begegnen, werden in MMDBMS zusätzlich unscharfe Anfragen benötigt. Damit sind alle diejenigen Verfahren und Methoden gemeint, die eine nicht hundertprozentige exakte Suche unterstützen. Häufig bedient man sich dazu verschiedener Methoden des Information Retrieval. Die nachfolgenden Ausführungen beschränken sich auf die im Zusammenhang mit Anfragen an MMDBMS wichtigen Methoden. Für eine umfangreiche Betrachtung des gesamten Gebietes des Information Retrieval sei an dieser Stelle auf [BR99] verwiesen. Da der Anfrageprozeß von

dem aus traditionellen DBMS bekannten abweicht, wird dieser nachfolgend erläutert. Anschließend werden die verschiedenen Möglichkeiten zur Formulierung von unscharfen Anfrageausdrücken in Multimedia-Anfragesprachen vorgestellt.

3.4.1 Der Anfrageprozeß

Um bei der Verwendung von unscharfen Anfragen in MMDBMS gute Suchergebnisse erzielen zu können, müssen im Vergleich zu konventionellen DBMS einige Änderungen im Anfrageprozeß vorgenommen werden. Da sich diese Änderungen auch auf die Anforderungen an eine Multimedia-Anfragesprache auswirken, wird der Anfrageprozeß in diesem Abschnitt grob beschrieben. Der Anfrageprozeß besteht aus den folgenden Schritten:

1. Anfragespezifikation,
2. Anfragebearbeitung und Optimierung,
3. Anfragebeantwortung und
4. Anfrage-Iteration.

Im ersten Schritt spezifiziert der Nutzer seine Anfrage. Die Anfragesprache sollte dazu exakte und unscharfe Prädikate unterstützen. Ähnlich wie in traditionellen DBMS wird die Anfrage im zweiten Schritt auf eine interne Darstellung abgebildet. Dabei sollte durch das MMDBMS eine Optimierung durch die Bestimmung eines optimalen Ausführungsplanes durchgeführt werden. Um bei einer unscharfen Anfrage entscheiden zu können, welche Objekte zur Ergebnismenge gehören, muß für jedes Objekt ein Relevanzwert bestimmt werden. Dieser läßt sich durch eine Distanzfunktion zwischen den Anfragemerkmalen und den Objektmerkmalen berechnen. Die Distanzfunktion liefert als Ergebnis einen Wert im Intervall $[0, 1]$, wobei der Wert „eins“ im allgemeinen bedeutet, daß ein Objekt bezüglich des Merkmals hundertprozentig relevant zur Anfrage ist. Ein vollkommen unrelevantes Objekt bekommt den Wert Null zugeordnet. In der Literatur lassen sich verschiedene Distanzfunktionen im Zusammenhang mit Indexverfahren für Multimedia-Daten finden [Fal99]. Auf eine ausführliche Betrachtung von Distanzfunktionen und Indexverfahren wird in dieser Arbeit verzichtet, da die Berechnung von Relevanzwerten Aufgabe des MMDBMS ist und somit für die Untersuchung von Anfragesprachen nur eine untergeordnete Rolle spielt.

Im dritten Schritt werden die Ergebnisse der Anfrage ausgegeben. In der Ergebnismenge sind dabei alle die Objekte enthalten, deren Relevanz einen bestimmten Schwellwert überschreitet. Dieser Schwellwert kann entweder von DBMS fest vorgegeben sein oder durch den Benutzer bei der Anfragespezifikation festgelegt werden. Die einzelnen Ergebnisobjekte sind dabei nach ihren Relevanzwerten in abnehmender Reihenfolge geordnet. Diese Ordnung wird im weiteren Verlauf der Arbeit auch als Ranking bezeichnet. In traditionellen DBMS ist der Anfrageprozeß an dieser Stelle beendet. Aufgrund der

Unschärfe in einer Anfrage kann aber nicht davon ausgegangen werden, daß die gefundenen Objekte immer sofort den Anforderungen des Benutzers entsprechen. Deshalb wird der Anfrageprozeß solange iterativ wiederholt, bis das Ergebnis den Nutzer befriedigt. In jedem Iterationsschritt versorgt der Nutzer das DBMS mit zusätzlichen Informationen, um die Anfrage zu verfeinern und die Relevanz der Ergebnisobjekte zu erhöhen. Dieser Iterationsprozeß bedarf einer speziellen Unterstützung durch die Anfragesprache, die durch traditionelle Sprachen, wie SQL oder OQL, im allgemeinen nicht gegeben ist. Inwieweit heutige Ansätze für eine Multimedia-Anfragesprache den Iterationprozeß unterstützen, wird im Zusammenhang mit der Beschreibung ausgewählter Ansätze in Kapitel 5 untersucht.

3.4.2 Unschärfe und Gewichtung in Anfrageausdrücken

Wie bereits erläutert, ist das Ergebnis einer unscharfen Anfrage eine nach Relevanzwerten geordnete Liste, wobei der Relevanzwert die Ähnlichkeit des Ergebnisobjektes bezüglich des Anfrageausdruckes angibt. Dabei stellt sich die Frage, wie man den Grad der Relevanz für die gesuchten Objekte durch einen Anfrageausdruck geeignet spezifizieren kann. In der Literatur [KB95, BCF99, Fag98] werden hierfür drei Möglichkeiten angegeben:

- durch die Verwendung von unpräzisen Ausdrücken und Prädikaten, wie zum Beispiel *normal*, *typisch* oder *kreisförmig* und *hauptsächlich blau* für Bildanfragen. Jeder dieser Ausdrücke stellt eine Menge möglicher Werte, aber keinen festen Einzelwert dar. Entsprechende quantitative Werte müssen diesen Ausdrücken vom DBMS zugeordnet werden.
- durch Prädikate, welche eine Ähnlichkeit oder Nähe zwischen Objekten ausdrücken. Dabei beschreiben die Prädikate keine präzise Beziehung zwischen Objekten oder zwischen Attributen bzw. Features und Werten. Die Suche nach Objekten basiert vielmehr auf der Berechnung einer semantischen Distanz zwischen den abgeleiteten Features des Anfrageobjektes und entsprechender Merkmale der Datenbankobjekte. Einige Beispiele für eine solche Suche sind:
 - die Ähnlichkeitssuche, das heißt, ein Objekt wird für die Suche nach zu diesem Objekt ähnlichen Objekten bezüglich bestimmter Merkmale in der Anfrage vorgegeben,
 - eine räumliche Umgebungssuche, also eine Suche anhand geographischer Beziehungen, zum Beispiel in Bildern oder Landkarten oder
 - die Freitextsuche, Suche nach Textdokumenten, in denen ein vorgegebener Wortlaut in ähnlicher Weise auftritt.
- durch die Zuordnung von Gewichten zu den einzelnen Anfragebedingungen, welche die Relevanz des Teilausdruckes bezüglich der gesamten Anfrage angeben. Im

allgemeinen liegen die Gewichte im Intervall $[0, 1]$, und die Summe aller Gewichte beträgt „eins“. In manchen Anfragesprachen werden auch Prozentangaben für die Festlegung der Relevanz einzelner Anfragebedingungen benutzt. Möglich ist auch die Verwendung unscharfer Ausdrücke, wie *hoch* oder *niedrig*. Folgendes Beispiel soll den Zusammenhang verdeutlichen: „Finde alle Bilder, auf denen ein Orchester (75%) und ein Dirigent (25%) abgebildet ist.“. Dabei werden zuerst alle Bilder gesucht, welche ein Orchester und einen Dirigenten abbilden. Sind von diesen Bildern nur wenige oder gar keine gefunden worden, so wird zusätzlich nach Bildern gesucht, die nur ein Orchester oder nur einen Dirigenten enthalten. Im Ranking der Ergebnismenge stehen an erster Stelle alle Bilder, die beides zeigen, ein Orchester und einen Dirigenten. Danach folgen alle Bilder, auf denen nur ein Orchester zu sehen ist, da diese ein höheres Gewicht (75%) in der Anfrage erhalten haben. Am Ende des Rankings sind dann alle Bilder zu finden, welche nur einen Dirigenten abbilden.

3.5 Präsentation von Anfrageergebnissen

In diesem Abschnitt werden die Besonderheiten beschrieben, welche bei der Präsentation der Anfrageergebnisse in einem MMDBMS beachtet werden müssen. Im Gegensatz zu traditionellen DBMS werden an ein MMDBMS weitaus höhere Präsentationsanforderungen gestellt. So beschränkt sich die Präsentation nicht nur auf einfach darzustellende Tabellen, sondern auch Bilder in verschiedenen Formaten und mit unterschiedlicher Qualität sowie Audio- und Videodokumente müssen dargestellt werden können. Besonders die Continuous-Medientypen, wie Audio und Video, bringen eine Reihe zeitlicher Abhängigkeiten mit sich, die durch das MMDBMS synchronisiert werden müssen. Nachfolgend sind die wichtigsten Merkmale einer Multimedia-Präsentation aufgeführt. Anschließend werden kurz einige Datenmodelle für die Präsentation von Multimedia-Daten vorgestellt.

3.5.1 Merkmale einer Multimedia-Präsentation

Dieser Unterabschnitt enthält eine Reihe von Merkmalen, die bei einer Multimedia-Präsentation von Bedeutung sind, und somit Anforderungen für die Präsentation der Anfrageergebnisse in MMDBMS darstellen. Diese sind im einzelnen:

- *Darstellung verschiedener Datentypen*, das heißt, für viele verschiedene Typen von Multimedia-Daten, wie Text, Bilder, Audio- und Videodaten, müssen Anzeige- und Abspielfunktionen angeboten werden.
- *Gerätespezifikation*, die Präsentation einzelner Ergebnisobjekte kann auf verschiedenen Geräten stattfinden. So kann ein Bild-Objekt in einem Fenster auf dem Monitor oder auf einem Drucker ausgegeben werden und das Abspielen einzelner Videos kann auf verschiedenen Bildschirmen erfolgen.

- *Medientransformation*, vor der Ausgabe werden eine Reihe von Merkmalen der Medienobjekte verändert. So kann eine Transformation in ein externes Darstellungsformat erfolgen. Zum Beispiel kann ein Bildobjekt im GIF- oder JPEG-Format ausgegeben werden. Außerdem kann die Qualität der Ausgabe anhand geeigneter Parameter (z.B. Größe und Farbtiefe für Bildobjekte) variiert werden. Das MMDBMS muß hierfür eine Grundmenge verbreiteter Ausgabeformate unterstützen. Ein Spezialfall der Medientransformation, welche einen Informationsverlust beinhalten, sind die sogenannten Filterfunktionen. Diese filtern bestimmte Merkmale für die weitere Verarbeitung aus den Multimedia-Daten heraus. Je nach Medientyp können dies beispielsweise Bildbearbeitungsfunktionen (Einfärben, Weichzeichnen usw.) oder Funktionen für die Audiosignalbearbeitung (Tonhöhenkorrektur, Klangänderungen) sein.
- *Medientranslation* ist der Übergang von einem Medium in ein anderes mit einem möglichst geringen Informationsverlust. Dies kann bei der Präsentation von Anfrageergebnissen durchaus sinnvoll sein. Ein Beispiel hierfür ist die Ausgabe eines Textes als Audio-Objekt oder die Darstellung der Schwingungskurve eines Audio-Objektes als Grafik.
- *Räumliche Constraints* müssen für die Präsentation visueller Mediendaten, wie Bild- und Video-Objekte, definiert werden können. Hierdurch läßt sich zum Beispiel festlegen, an welcher Position auf dem Monitor ein Bild angezeigt werden soll.
- *Zeitliche Constraints und Synchronisation* resultieren aus den zeitlichen Abhängigkeiten von und zwischen den Mediendaten. Speziell die Continuous-Mediendaten stellen bei der Präsentation hohe Anforderungen an das MMDBMS, da der Benutzer ein unterbrechungsfreies Abspielen dieser Daten erwartet. Weitere Synchronisationsanforderungen entstehen dadurch, daß eine Multimedia-Präsentation aus mehreren Medienobjekten besteht, welche zu bestimmten, zuvor festgelegten Zeitpunkten angezeigt bzw. abgespielt werden müssen. Das MMDBMS hat dabei für die Synchronisation der Darstellung dieser Medienobjekte zu sorgen.
- *Benutzerinteraktionen* können während einer Multimedia-Präsentation eintreten und müssen vom System entsprechend verarbeitet werden. Möglich wäre zum Beispiel die Änderung der Lautstärke beim Abspielen eines Audio-Objektes. Für die Interaktion bei der Darstellung eines Video-Objektes werden im allgemeinen Funktionen wie schneller Vor- und Rücklauf, Pause und Play angeboten.

3.5.2 Datenmodelle für die Präsentation

Für die Beschreibung der räumlichen und zeitlichen Constraints können im wesentlichen die in den Abschnitten 3.2.2 und 3.3.1 vorgestellten räumlichen und zeitlichen Beziehungen genutzt werden. Die räumliche Anordnung einzelner Medienobjekte kann dabei

sowohl durch topologische und Richtungsbeziehungen erfolgen als auch durch die Angabe fester Bildschirmkoordinaten. Da jedem Multimediaobjekt, das präsentiert werden soll, ein festes Zeitintervall zugeordnet ist, können die von Allen [All83] eingeführten 13 Zeitintervallbeziehungen zur Formulierung der zeitlichen Constraints genutzt werden.

Eine Reihe weiterer Parameter, welche die geforderte Mindestqualität für die Präsentation einzelner Objekte festlegen, sind mit den Medienobjekten verknüpft. Diese Parameter werden auch als Quality-of-Service Parameter (QoS) bezeichnet. Sie werden benötigt, um Präsentationen in unterschiedlicher Qualität, je nach Anforderung des Benutzers, mit möglichst geringer Systembelastung zu erstellen. Die Auswahl geeigneter QoS-Parameter und die Verwendung dieser Parameter zur Erstellung einer optimalen Präsentation sind Gegenstand aktueller Forschungsarbeiten, auf die an dieser Stelle verwiesen sei [Nah99, Sha99].

Als Datenmodelle für die Präsentation werden im wesentlichen die schon in Abschnitt 3.3 vorgestellten Modelle verwendet. Sowohl zeitliche feste Modelle, wie das Timeline-Diagramm, als auch zeitlich flexible Präsentationsgraphen, meist dargestellt durch Petrinetze, kommen hierbei zum Einsatz.

4 | Anforderungen an Multimedia-Anfragesprachen

Eine Reihe von unterschiedlichen Anfragesprachen für Multimedia-Daten wurde in den vergangenen Jahren vorgeschlagen. Erweiterungen der relationalen Datenbank-Anfragesprache SQL zählen ebenso dazu, wie objektorientierte Ansätze, beispielsweise OQL-Erweiterungen oder graphische Anfragesprachen. Als Grundlage für einen Vergleich dieser verschiedenen Ansätze werden Kriterien benötigt, welche unabhängig von einem konkreten Datenmodell sind und dabei die Qualität einer Sprache wiedergeben. In der Literatur [SST97, Heu97, LV97] lassen sich verschiedene Vorschläge einer Kriterienliste für Anfragesprachen finden. Basierend auf einer recht ausführlichen Auflistung von Kriterien in [SST97] sind im nachfolgenden Abschnitt allgemeine Anforderungen an Anfragesprachen aufgeführt. Dabei wird besonderes Augenmerk auf die Gültigkeit der einzelnen Kriterien für den besonderen Fall der Multimedia-Datenbanken gelegt.

In einem generellen Kriterienkatalog finden die speziellen Anforderungen an Anfragesprachen für MMDBMS natürlich keine Beachtung. Abbildung von räumlichen und zeitlichen Beziehungen in Anfragen, die geeignete Präsentation der Anfrageergebnisse und die Möglichkeit, unscharfe Anfragen zu formulieren, sind nur einige dieser speziellen Anforderungen. Eine ausführliche Betrachtung der Besonderheiten und der daraus abgeleiteten speziellen Kriterien für Multimedia-Anfragesprachen erfolgt in Abschnitt 4.2.

4.1 Allgemeine Anforderungen an Anfragesprachen

Multimedia-Anfragesprachen müssen dieselben allgemeinen Anforderungen erfüllen, wie Anfragesprachen relationaler oder objektorientierter Datenbanken. Deshalb sind nachfolgend die wesentlichen Kriterien aufgeführt und werden in Hinblick auf Multimedia-Datenbanken näher untersucht. Zu beachten ist, daß diese Kriterien nicht unabhängig voneinander sind, sondern teilweise in Beziehung stehen oder sich widersprechen. Auf diese Besonderheiten wird an den entsprechenden Stellen eingegangen. Eine spezielle Anfragesprache kann somit nicht alle nachfolgenden Kriterien gleich gut erfüllen. Weiterhin muß die Gültigkeit der Kriterien für Anfragesprachen an Multimedia-Datenbanken untersucht werden. Die nachfolgenden Unterabschnitte gliedern sich jeweils in zwei Teile. Der erste Teil erläutert das jeweilige Kriterium allgemein, der zweite Teil bewertet dieses bezüglich der Anfragesprachen in MMDBMS und stellt dazu konkrete Anforderungen auf.

Ad-hoc-Formulierung

Neben dem Zugriff auf die Datenbank durch Anwendungsprogramme muß ein DBMS die Möglichkeit bieten, Anfragen direkt über eine interaktive Benutzerschnittstelle zu formulieren. Die Anfragesprache sollte dazu so entworfen sein, daß kein vollständiges Anwendungsprogramm geschrieben werden muß, um auf die Daten zuzugreifen. Diese Art der direkten Formulierbarkeit von Anfragen wird Ad-hoc-Formulierung genannt. Teilweise besteht ein Zusammenhang zur weiter unten aufgeführten Deskriptivität.

Eine Ad-hoc-Formulierung sollte auch von einem MMDBMS unterstützt werden. Dabei ist es nötig, neben textbasierten Konstrukten auch verschiedene Medientypen als Teil der Anfrage einzubinden. Die Anfragesprache muß hierfür geeignete Methoden anbieten, wie zum Beispiel eine grafische Form der Anfrageformulierung zur Spezifikation räumlicher Beziehungen. In diesem Zusammenhang sollte auch die Einbeziehung von verschiedenen Geräten zur Eingabe der Medienobjekte, z.B. Scanner für Bilder, möglich sein. Des weiteren werden für die Anfragesprache Basisfunktionalitäten zur Spezifikation der Ergebnispräsentation einer Ad-hoc-Anfrage benötigt.

Generische Operationen

Die Unterstützung von generischen Operatoren durch die Anfragesprache bedeutet, daß für semantisch gleiche Operationen auf verschiedenen Datenbankobjekten die gleiche Syntax verwendet wird. Somit muß ein Anwender nur wenige Sprachkonstrukte erlernen, um eine Anfragesprache zu beherrschen. Zusätzlich unterstützen generische Operationen durch Abstraktion von den Implementierungsdetails und der Anwendungssemantik die Datenunabhängigkeit.

Eine Anfragesprache für Multimedia-Datenbanken sollte in generischen Operationen auch Medienobjekte unterstützen, beispielsweise für die Selektion, Projektion und verschiedene Mengenoperationen auf den Medienobjekten. Des weiteren ist es sinnvoll, sowohl für die Beschreibung der räumlichen und zeitlichen Beziehung zwischen Medienobjekten innerhalb einer Anfrage als auch für die Beschreibung der Präsentation der Anfrageergebnisse einheitliche Attribute zur Formulierung der Raum-/Zeitzusammenhänge in Anfragen (vgl. Abschnitt 3.2.2) zu verwenden. Dasselbe gilt für Spatial- und Temporal-Joins, welche wie normale Joins durch die Anfragesprache unterstützt werden sollten.

Anwendungsunabhängigkeit

Der Entwurf einer Anfragesprache darf nicht im Hinblick auf einen bestimmten Zweck oder eine spezielle Anwendung erfolgen, er sollte vielmehr eine universelle Verwendung unterstützen. Diese Kernforderung entkoppelt das DBMS von den Anwendungen. Somit lassen sich ohne gleichzeitige Beeinflussung des DBMS neue Anwendungen hinzufügen oder alte Anwendungen ändern bzw. entfernen. Das DBMS dient hierbei als zentrales System für viele verschiedene Anwendungen. Die Anwendungsunabhängigkeit ist eine

wichtige Forderung des Prinzips der Datenunabhängigkeit, welche für eine langfristige Benutzung der Datenbestände durch verschiedene Anwendungen von grundlegender Bedeutung ist. Erreicht wird die Datenunabhängigkeit durch die Drei-Ebenen-Architektur und die damit verbundene Unterteilung in internes Schemata, konzeptuelles Schemata und externes Schemata [HS97]. Auch um diese heute im allgemeinen akzeptierte Architektur zu unterstützen, sollte eine Anfragesprache soweit wie möglich anwendungsunabhängig sein.

Für Multimedia-Anfragesprachen kann die Forderung nach Anwendungsunabhängigkeit uneingeschränkt übernommen werden. Gewünscht ist dabei eine Unabhängigkeit von konkreten Medienformaten und Kompressionsverfahren. Die Möglichkeit der Transformation von einem Medium in ein anderes (z.B Text als Audiodokument ausgegeben) sollte bestehen und von der Anfragesprache unterstützt werden.

Deskriptivität

Der Benutzer eines DBMS soll beim Formulieren seiner Anfrage die Struktur und den Inhalt der Datenbankobjekte, welche er sucht, beschreiben. Durch die Anfragesprache dürfen keine Berechnungsalgorithmen oder eine Abarbeitungsreihenfolge zur Anfragebearbeitung vorgegeben werden. Diese als Deskriptivität bezeichnete Eigenschaft einer Anfragesprache sichert die Datenunabhängigkeit und ist Voraussetzung für eine Optimierung durch das DBMS.

Jede Datenbankanfragesprache, egal ob für Multimedia-, relationale- oder Objektdatenbanken, sollte durch ein hohes Maß an Deskriptivität charakterisiert sein, um dem Anfrageprozeß Potential für eine Optimierung zu lassen und unabhängig gegenüber Änderungen der Implementierung zu sein. Auch im Zusammenhang mit der Präsentation von Anfrageergebnissen muß eine Multimedia-Anfragesprache die Deskriptivität beachten. So sollten die Konstrukte der Anfragesprache zur Präsentationsbeschreibung ebenfalls nur den Inhalt und die Struktur der Präsentation beschreiben und nicht die Art und Weise, wie diese zu erstellen ist.

Mengenorientiertheit

Das Ergebnis einer Anfrage in den meisten DBMS ist eine Menge von Objekten. Da innerhalb einer Menge normalerweise keine Ordnung besteht, kann das DBMS die Elemente in einer beliebigen Reihenfolge zurückliefern. Dies kann unmittelbar zur Optimierung bei der Anfragebearbeitung ausgenutzt werden und führt zu einer höheren Effizienz im Anfrageprozeß.

Die Mengenorientiertheit ist auch bei Multimedia-Anfragesprachen eine wünschenswerte Eigenschaft, es müssen aber einige Besonderheiten beachtet werden. In einem MMDBMS ist es sinnvoll, für die Suche nach komplexen oder schwer beschreibbaren Datenbankobjekten Ähnlichkeitsanfragen zuzulassen. Diese in Abschnitt 3.4 ausführlich beschriebenen Anfragen liefern keine exakten Ergebnisse. Die Ergebnismenge besteht

vielmehr aus Datenbankobjekten, welche ähnliche Eigenschaften aufweisen wie sie in der Anfrage formuliert wurden. Als Maß für die Ähnlichkeit zwischen den Ergebnisobjekten und der Anfrage wird eine Gewichtung angegeben, welche, bezogen auf das gesamte Ergebnis, als Ranking bezeichnet wird. Dieses Ranking ist eine geordnete Liste, deren Reihenfolge im Anfrageprozeß beachtet werden muß. Die einzelnen Mengenoperationen, wie Vereinigung, Durchschnitt, Differenz usw., sollten hierfür um Konzepte zur Behandlung von Relevanzwerten erweitert werden. Beim Entwurf der Multimedia-Anfragesprache muß die Semantik dieser Mengenoperationen eindeutig festgelegt werden. Die Präsentation der Anfrageergebnisse unscharfer Anfragen sollte in Form einer nach den Relevanzwerten geordneten Liste erfolgen.

Orthogonalität

Wichtig für komplexe Anfragen ist die Möglichkeit der Kombination von Anfragekonstrukten. Dabei kann innerhalb einer Anfrage ein beliebiges Konstrukt durch ein anderes ersetzt werden, solange der Ergebnistyp des eingesetzten Konstruktes zum Ergebnistyp des zu ersetzenden Konstruktes paßt. Diese als Orthogonalität bezeichnete Eigenschaft von Anfragesprachen ermöglicht unter anderem das Schachteln von Anfragen und bildet damit die Voraussetzung für komplexe Anfragekonstrukte.

MMDBMS basieren ähnlich wie ODBMS auf einem Typsystem. Dieses enthält neben einfachen Datentypen (wie z.B. Character) und allgemeinen Objekttypen auch Medientypen, beispielsweise Bilder und Videosequenzen. Betrachtet man ein Konstrukt einer Anfrage als Ausdruck eines bestimmten Typs, so ist eine Ersetzung dieses Konstruktes möglich, wenn die Datentypen zueinander passen. Eine Multimedia-Anfragesprache sollte diese Eigenschaft ausnutzen und, solange die Typübereinstimmung gewährleistet ist, ein beliebiges Ersetzen von Konstrukten in Anfragen erlauben, egal, ob es sich um einfache Datentypen oder Medientypen handelt.

Effizienz

Ein Benutzer erwartet von einem DBMS kurze Antwortzeiten auf seine Anfragen. Um dies zu gewährleisten, müssen die durch eine Anfragesprache zur Verfügung gestellten Konstrukte effizient zu realisieren sein. Das heißt, die Ergebnismenge kann mit einem akzeptablen Aufwand (maximal $O(n^2)$, n Anzahl der betrachteten Objekte) für die Basisstrukturen berechnet werden. Eine Unterstützung von komplexen Basisfunktionen durch die Anfragesprache, deren Berechnung einen quadratischen Aufwand wesentlich überschreitet, kann im Hinblick auf die Performanz des gesamten DBMS als nicht sinnvoll erachtet werden.

Ein Effizienzproblem ergibt sich in ODBMS durch den Einsatz von benutzerdefinierten Methoden. Für diese kann keine Aufwandsgrenze garantiert werden und somit auch nicht für die Anfragen, welche diese Methoden verwenden. Dieses Problem bleibt auch für MMDBMS bestehen, die auf Objektdatenbanken aufbauen. Hinzu kommen

Effizienzprobleme durch das hohe Datenvolumen in MMDBMS und durch komplexe Algorithmen für Kompressionsverfahren und Ähnliches. Deshalb muß man sich mit der Forderung nach effizient berechenbaren Anfragekonstrukten auf die einfachen Operationen beschränken. In einem MMDBMS kann das Ergebnis einer Anfrage ein langer oder endloser Medienstream (Video- oder Audiostream) sein. Eine Ausgabe dieses Streams in der bestmöglichen Qualität stellt sehr hohe Anforderungen an die Leistungsfähigkeit des MMDBMS und wird in vielen Fällen (z.B. zur Vorschau) nicht benötigt. Um eine hohe Effizienz des Gesamtsystems zu erreichen, können deshalb für die Präsentation eines Medienstreams verschiedene Parameter zur Festlegung der Ausgabequalität (vgl. Abschnitt 3.5.1) vorgegeben werden. Durch die Anfragesprache muß die Spezifikation solcher Parameter unterstützt werden.

Erweiterbarkeit

Datenbank-Management-Systeme lassen sich um zusätzliche Typen und Methoden erweitern. Diese Erweiterungen müssen durch eine Anfragesprache unterstützt werden.

Die Erweiterbarkeit ist in bezug auf MMDBMS eine besonders wichtige Forderung, da sich somit das Typsystem um neue Medientypen erweitern läßt. Für spezielle Anwendungen müssen spezielle Features abgeleitet und in der Datenbank abgelegt werden können. Eine Multimedia-Anfragesprache sollte sich leicht an diese Erweiterungen anpassen lassen bzw. die Möglichkeit bieten, auf die Erweiterungen im Typsystem und auf zusätzliche Features in Anfragen zu zugreifen.

Abgeschlossenheit

Das Kriterium der Abgeschlossenheit fordert, daß jedes Ergebnis einer Anfrage bzw. eines Anfragekonstruktes in dem zugrunde liegenden Datenmodell darstellbar ist. Dies sichert, daß das Ergebnis einer Anfrage als Eingabe für eine nachfolgende Anfrage genutzt werden kann.

Ein Grundproblem von ODBMS ist die Frage nach der Semantik der Anfrageergebnisse. Diese stellt sich auch für die auf dem OID-Konzept basierende MMDBMS. Dabei sind die drei folgenden Ansätze für eine Semantikfestlegung möglich:

- relationale Semantik: komplexer Wert als Anfrageergebnis,
- objektgenerierende Anfragen: neue Objekte als Anfrageergebnis,
- objekterhaltende Anfragen: aus existierenden Objekten zusammengesetzte Anfrageergebnisse.

Alle drei Ansätze haben ihre Berechtigung im entsprechenden Kontext. Eine Entscheidung für eine relationale oder objektgenerierende Semantik wird entweder explizit bei der Anfrageformulierung getroffen oder bereits durch das Datenmodell festgelegt. Weiterhin hängt die Auswahl vom Typ der Anfrage ab, z.B. ob komplexe Werte berechnet

werden oder nur Mengen von Objekten auszuwählen sind. Für eine ausführliche Betrachtung und einen Vergleich dieser Ansätze, welche bereits im Zusammenhang mit ODBMS untersucht wurden, sei auf [SST97] verwiesen. Anfragesprachen in MMDBMS müssen neben den oben beschriebenen Zusammenhängen zusätzlich die Semantik von Medienobjekten und Relevanzwerten als Anfrageergebnisse festlegen. Relevanzwerte lassen sich als zusätzliches Attribut den Ergebnisobjekten zuweisen, welche geordnet nach ihren Relevanzwerten als Liste dargestellt werden können. Für Medienobjekte ist sowohl eine objektgenerierende als auch eine objekterhaltene Semantik möglich, was entweder durch die Anfragesprache oder durch das Datenmodell eindeutig festgelegt werden sollte.

Adäquatheit

Die Adäquatheit verlangt, daß alle Konstrukte des Datenbankmodells durch die Anfragesprache ausgenutzt werden können. Somit ist die Adäquatheit das Gegenstück zur Abgeschlossenheit. Beide Kriterien zusammen verlangen von einer Anfragesprache, daß diese das gesamte Datenmodell ausnutzt, aber nur dieses.

Insbesondere für Anfragesprachen an MMDBMS verlangt die Adäquatheit, daß die räumlichen und zeitlichen Beziehungen der Multimedia-Daten und die zugeordneten Metadaten zur Formulierung von Anfragen benutzt werden können. Des weiteren muß eine Anfragesprache Konstrukte zum Formulieren von unscharfen Anfragen und die Präsentation der Anfrageergebnisse aufweisen. Sie sollte medienübergreifende Anfragen unterstützen. Wenn im Einzelfall das Datenmodell spezielle Inhaltsbeschreibungen der Multimedia-Dokumente abbildet, wie zum Beispiel Angaben zur Gestalt oder Farbverteilung eines Bildes, müssen sich diese Inhaltsbeschreibungen auch innerhalb einer Anfrage ausdrücken lassen, um das Kriterium der Adäquatheit zu erfüllen.

Vollständigkeit

Das Kriterium der Vollständigkeit ist eine Verallgemeinerung der relationalen Vollständigkeit. Diese fordert von relationalen Anfragesprachen die gleiche Ausdrucksfähigkeit, wie sie die Relationenalgebra bietet. Daran angelehnt drückt das Kriterium der Vollständigkeit im allgemeinen eine Gleichmächtigkeit zu einer Standardsprache des jeweiligen Datenmodells aus. Beispielsweise wird von Anfragesprachen für ODBMS mindestens die Gleichmächtigkeit zu den objektorientierten Äquivalenten der Operationen der Relationenalgebra gefordert.

Daraus läßt sich schließen, daß eine Multimedia-Anfragesprache entsprechend dem zugrunde liegenden Datenmodell mindestens die Mächtigkeit der Relationenalgebra bzw. dem objektorientierten Äquivalent der Relationenalgebra besitzen sollte. Wenn in Zukunft eine Algebra speziell für Multimedia-Datenbanken entwickelt würde, was in Anbetracht der speziellen Eigenschaften von Multimediaobjekten durchaus sinnvoll erscheint, müßte von einer Anfragesprache eine Gleichmächtigkeit zu dieser Algebra gefordert werden, um das Kriterium der Vollständigkeit zu erfüllen.

Sicherheit

Die Terminierung und die Berechnung eines endlichen Ergebnisses für eine syntaktisch korrekt formulierte Anfrage sind die wesentlichen Anforderungen, welche unter dem Begriff der Sicherheit zusammengefaßt sind. Um diese Sicherheit garantieren zu können, darf die Anfragesprache somit nicht berechnungsvollständig sein, da sonst Endlosschleifen o. Ä., formuliert werden könnten.

Besonders problematisch in diesem Zusammenhang sind Methodenaufrufe in Anfragesprachen für ODBMS und für MMDBMS. Bei einer berechnungsvollständigen Methodensprache liegt die Sicherheit in der Hand des Programmierers, welcher die Methoden entwirft. Dies ist natürlich nicht erwünscht. Deshalb werden in der Literatur drei Alternativen vorgeschlagen:

1. keine Methodenaufrufe in Anfragesprachen,
2. Zusicherungsverfahren für benutzerdefinierte Methoden oder
3. Verwendung einer entscheidbaren Teilsprache.

Verbietet man Methodenaufrufe in Anfragesprachen, widerspricht dies der Adäquatheit, da Methodenaufrufe eines der Grundprinzipien des objektorientierten Ansatzes sind. Des weiteren werden in MMDBMS zusätzliche Methoden für einzelne Anwendungen benötigt. Zum Beispiel müssen für Gesichtserkennungssysteme spezielle Merkmale des Gesichtes erkannt und abgespeichert werden. Dazu braucht man spezielle Methoden, auf die auch in Anfragen zugegriffen werden kann. Der Ansatz, welcher den Aufruf selbstdefinierter Methoden in Anfragesprachen verbietet, ist somit für MMDBMS ungeeignet. Wird für selbstdefinierte Methoden ein Zusicherungsverfahren verwendet, muß der Programmierer eine bestimmte Komplexität und die Terminierung seiner Methoden sicherstellen, damit diese vom MMDBMS akzeptiert werden. Somit können selbstdefinierte Methoden in Anfragen benutzt werden. Die Verantwortung für die Sicherheit liegt aber weiterhin größtenteils beim Programmierer. Eine entscheidbare Teilsprache hingegen läßt nur überprüfbare Konstrukte für die Methodenprogrammierung zu. Hier können unter Umständen bestimmte Funktionalitäten nur schwer oder gar nicht umgesetzt werden. Eine Anfragesprache für MMDBMS sollte eine der beiden Methoden, Zusicherung oder Verwendung einer berechnungsunvollständigen Teilsprache, unterstützen.

Optimierbarkeit

Aus der Deskriptivität ergibt sich die Forderung nach Optimierbarkeit. Für Anfragen, welche sich unabhängig von Speicherstrukturen und Implementierungsdetails formulieren lassen, hat das DBMS im allgemeinen die Aufgabe, sie zu optimieren. Von einer Anfragesprache wird für diesen Zweck erwartet, daß sie sich auf ein internes Anfragemodell abbilden läßt.

Das Kriterium der Optimierbarkeit muß von Anfragesprachen an MMDBMS ebenso erfüllt werden.

Formale Semantik

Damit sich durch das DBMS die oben beschriebene Optimierung durchführen läßt, muß eine Anfrage auf eine interne Anfragedarstellung abgebildet werden können, welche die komplexen Zusammenhänge zwischen Betriebssystem, den Ein- und Ausgabegeräten, der Pufferverwaltung und der Synchronisation berücksichtigt. Um diese Abbildung vornehmen zu können, ist ein geeigneter Äquivalenzbegriff und damit verbunden eine formale Semantik notwendig. Eine formale Semantik ist wichtig, um die Bedeutung eines Anfragerterms eindeutig festzulegen.

Die Forderung nach einer formalen Semantik für MMDBMS ist somit selbstverständlich, wurde aber in den meisten Ansätzen für Anfragesprachen in MMDBMS bislang nur sehr oberflächlich und unvollständig behandelt. Einige semantische Aspekte wurden bereits im Zusammenhang mit der Sicherheit einer Anfrage betrachtet.

Eingeschränktheit

Die Eingeschränktheit faßt im wesentlichen zusammen, was schon mehrfach diskutiert wurde. Eine Anfragesprache kann unter Berücksichtigung von Effizienz, Optimierbarkeit und Sicherheit nicht berechnungsvollständig sein.

Aktuelle Trends in der Datenbankentwicklung widersprechen teilweise der Forderung nach Eingeschränktheit, um die Fähigkeiten des MMDBMS zu erweitern. In diesem Fall muß genau untersucht werden, ob diese Erweiterungen einen Verzicht auf die Eingeschränktheit rechtfertigen. Zumindest für den Sprachkern sollte aus den oben genannten Gründen die Forderung nach Eingeschränktheit aufrecht erhalten werden.

4.2 Spezielle Anforderungen an Multimedia-Anfragesprachen

Im vorhergehenden Abschnitt wurden Kriterien aufgestellt, welche allgemein für Anfragesprachen gelten, egal ob die Sprache für ein RDBMS, ODBMS oder MMDBMS entworfen wurde. Zusätzlich ergeben sich für den speziellen Bereich der MMDBMS aufgrund der besonderen Eigenschaften der Multimedia-Daten weitere Anforderungen, denen eine Multimedia-Anfragesprache gerecht werden sollte. So müssen die umfangreichen räumlichen und zeitlichen Beziehungen, die innerhalb und zwischen den Multimediaobjekten bestehen, beachtet werden. Des weiteren lassen sich mit traditionellen Anfragesprachen nur exakte Anfragen auf konventionellen Daten formulieren. Dies mag für Anfragen an Metadaten und Beschreibungen von Multimedia-Daten ausreichen, aber für eine inhaltsbasierte Suche mit nicht exakten Anfragen werden spezielle Anfragekonstrukte, die über den traditionellen Ansatz hinausgehen, benötigt. Wie sich leicht erkennen läßt, ergeben sich einige spezielle Anforderungen für Anfragesprachen an MMDBMS, welche in diesem Abschnitt in Form von weiteren Kriterien aufgelistet und erläutert werden.

Universell

Viele der heutigen Ansätze für Multimedia-Anfragesprachen wurden im Hinblick auf eine bestimmte Anwendung (z.B. medizinische Bilddatenbanken) entworfen oder konzentrieren sich auf einzelne Medientypen (z.B. Videodatenbanken). Beides sind keine wünschenswerten Eigenschaften für eine universelle Anfragesprache. Die Anwendungsunabhängigkeit wurde bereits als allgemeine Anforderung besprochen. Von einer universellen Multimedia-Anfragesprache wird zusätzlich erwartet, daß diese mindestens die sogenannten „VITA“-Datentypen (Video-, Image-, Text-, Audiodaten) unterstützt. Des weiteren sollten sich die Anfragen nicht auf einzelne Medientypen beschränken, sondern vielmehr medienübergreifend formulierbar sein.

Inhaltsbasierte Anfragen

Techniken, die eine Suche nach Multimediaobjekten, basierend auf dem Inhalt der Objekte, ermöglichen, stellen eine Grundanforderung an jedes MMDBMS dar. Die Unterstützung inhaltsbasierter Anfragen ist somit eine Kernforderung für jede Multimedia-Anfragesprache. Dazu müssen Anfragen sowohl durch inhaltsbeschreibende Attribute als auch durch Featurewerte, abgeleitet aus den Multimedia-Daten, spezifiziert und zu komplexen Anfragen zusammengesetzt werden können. Des weiteren sollten Anfragen an die Struktur der Multimediaobjekte formulierbar sein.

Eine weitere Anforderung besteht in der Unterstützung der Abschwächung bzw. Verschärfung von Anfragen durch spezielle Anfragekonstrukte. Durch diese Anfragekonstrukte sollte sich spezifizieren lassen, ob und wie stark eine Abschwächung/Verschärfung einzelner Anfragekonstrukte gewünscht ist.

Da die aus den Multimedia-Daten abgeleiteten Inhaltsinformationen teilweise sehr unpräzise sind, werden für eine leistungsfähige inhaltsbasierte Suche ebenfalls unscharfe Anfragen benötigt. Die Anforderungen, welche nachfolgend im Zusammenhang mit der Unterstützung unscharfer Anfragen aufgestellt werden, gelten somit ebenso für inhaltsbasierte Anfragen.

Räumliche Anfragen

Die räumlichen Eigenschaften von Multimedia-Daten wurden bereits im Zusammenhang mit der Klassifikation von Medientypen und der Beschreibung von räumlichen Anfragen in Kapitel 2 diskutiert. Dabei wurden 12 Richtungsbeziehungen und 6 topologische Beziehungen vorgestellt. In einer Multimedia-Anfragesprache sollten diese als Prädikate in der „where“-Klausel verwendbar sein, um dadurch räumliche Zusammenhänge in der Anfrage ausdrücken zu können. Des weiteren müssen Operationen auf räumlichen Daten unterstützen werden. Wesentliche Grundoperationen wurden dazu in Abschnitt 3.2 in Kapitel 2 vorgestellt. Zusätzlich muß die Anfragesprache auch weitere vom Anwendungsprogrammierer entworfene Funktionen integrieren können. Dies wurde bereits im Zusammenhang mit dem Kriterium der Sicherheit im vorhergehenden Abschnitt

erläutert. Dabei muß an verschiedenen Stellen der Anfrage ein Funktionsaufruf möglich sein, zum Beispiel:

- als Aggregatfunktion,
- zum Ableiten bestimmter Eigenschaften der Ergebnisobjekte oder
- zur Berechnung komplexer Prädikate.

Nachfolgend sind einige aus Sicht des Anwenders typische Anfragen an räumliche Daten aufgeführt. Die verschiedenen Anfragen sollten durch die Multimedia-Anfragesprache abgedeckt werden, um eine komplexe Suche auf räumlichen Daten zu ermöglichen. Diese sind im einzelnen:

1. Unterstützung einer direkten räumlichen Suche: das heißt, bestimmte Objekte werden in vorgegebenen Bildregionen gesucht. Ein Beispiel für eine solche Anfrage wäre: „Finde alle Autos im rechten unteren Teil des Videoframes.“
2. Eine kombinierte räumliche Suche: dabei erfolgt die Suche nach Objekten durch bestimmte Attribute und Beziehungen zwischen diesen Attributen und den räumlichen Objekten, zum Beispiel: „Zeige den Typ, das Baujahr, den Kilometerstand und ein Foto des Autos, welches rechts unten auf dem Bild zu sehen ist.“
3. Unterstützung einer komplexen räumlichen Suche durch Ausnutzung von Mengenoperationen über räumliche Attribute: „Finde alle Straßen, welche sich im Punkt (x,y) schneiden.“
4. Direkte Berechnung spezieller räumlicher Aggregatfunktionen von Objekten mit räumlichen Eigenschaften: „Berechne die Gesamtfläche aller auf dem Bild abgebildeten Wälder.“
5. Zusammengesetzte räumliche und zeitbezogene Suche: das heißt, räumliche und zeitliche Beziehungen werden zur Suche in den Daten ausgenutzt, wie zum Beispiel in der folgenden Anfrage: „Finde die Stellen im Video, wo eine Person neben einem Auto steht und anschließend in diesem Auto sitzt.“

Zeitliche Anfragen

Die Abhängigkeit von der Zeit ist ein wesentliches Merkmal vieler Multimedia-Daten, wie schon in Abschnitt 3.3 festgestellt wurde. Aus der Forderung nach Adäquatheit, nach welcher alle Konstrukte des Datenmodells durch eine Anfragesprache unterstützt werden sollen, folgt, daß eine Multimedia-Anfragesprache die Formulierung zeitlicher Anfragen unterstützen muß. Eine typische zeitbezogene Anfrage wäre: „Finde alle Personen in einem Überwachungsvideo, welche die Bank betreten haben, bevor der Überfall stattfand.“ Die Spezifikation des Prädikats „bevor“ benötigt dabei eine spezielle Unterstützung

durch die Anfragesprache. Eine Multimedia-Anfragesprache muß zur Unterstützung zeitbezogener Anfragen in der Lage sein, verschiedene zeitliche Zusammenhänge und Operationen abzubilden, wie sie beispielsweise in Kapitel 2 diskutiert wurden. Diese sind nachfolgend noch einmal zusammengefaßt:

- Selektion bestimmter Zeitpunkte, Zeitintervalle und Zeitspannen,
- Ausnutzung von zeitlichen Beziehungen zwischen den Multimediaobjekten zur Anfrageformulierung,
- Einbindung von Operationen auf Zeitintervallen, Zeitspannen und Zeitpunkten in die Anfragesprache.

Des weiteren muß sich eine Anfragesprache für MMDBMS um zusätzliche zeitbezogene Prädikate und Operationen erweitern lassen, falls diese für spezielle Anwendungsgebiete benötigt werden.

Unschärfe Anfragen

Wie im Zusammenhang mit unscharfen Anfragen in Abschnitt 3.4 und der Inhaltsbeschreibung durch Metadaten in Abschnitt 2.2 ausführlich erläutert wurde, ist eine exakte Beschreibung der Mediendaten nicht möglich. Für eine leistungsfähige Suche in einem MMDBMS müssen deshalb unscharfe Anfragen formuliert werden können. Dies bedarf der Unterstützung durch die Multimedia-Anfragesprache. Folgende Möglichkeiten zur Beschreibung von Unschärfe und Nähe und zur Gewichtung von Anfrageausdrücken sollte eine Multimedia-Anfragesprache anbieten:

- unscharfe Prädikate (z.B. normal, kreisförmig, hauptsächlich blau), die einen Bereich möglicher Werte und keinen einzelnen exakten Wert vorgeben,
- Ähnlichkeitsanfragen, das heißt, anhand eines in der Anfrage spezifizierten Multimediaobjektes und einer Distanzfunktion wird nach Objekten gesucht, die eine bestimmte Nähe oder Ähnlichkeit zum vorgegebenen Multimediaobjekt bezüglich der Distanzfunktion aufweisen, und
- gewichtete Anfrageausdrücke, wobei den einzelnen Anfragebedingungen Gewichte entsprechend ihrer Relevanz bezüglich der gesamten Anfrage zugeordnet werden. Die Anfragesprache sollte dazu entweder die Zuordnung von Zahlenwerten (z.B. 0,75) oder die Zuordnung von unscharfen Ausdrücken (z.B. hoch, niedrig) unterstützen.

Weiterhin sollte ein iterativer Anfrageprozeß, wie er in Abschnitt 3.4.1 beschrieben wurde, durch die Anfragesprache unterstützt werden, um so die Qualität der Suchergebnisse schrittweise erhöhen zu können. Wie bereits im Zusammenhang mit der Forderung nach Mengenorientiertheit erläutert wurde, müssen die Mengenoperationen um Konzepte zur

Behandlung der den Ergebnisobjekten unscharfer Anfragen zugeordneten Relevanzwerten erweitert werden. Beim Entwurf einer Multimedia-Anfragesprache muß hierfür eine geeignete Semantik festgelegt werden.

Präsentation von Anfrageergebnissen

Die Anfragesprache spielt bei der Präsentation der Anfrageergebnisse eine wichtige Rolle. Ihre Aufgabe ist es, alle Parameter und Funktionen zu spezifizieren, die für eine Präsentation benötigt werden. Die genaue Menge von zu unterstützenden Parametern und Funktionen hängt hierbei von der Art der Anfragestellung ab. Dabei muß unterschieden werden zwischen:

1. Formulierung einer Anfrage durch ein Anwendungsprogramm oder
2. Formulierung einer Anfrage über die interaktive Benutzerschnittstelle.

Im ersten Fall werden die Anfragen durch ein Anwendungsprogramm gestellt. Die zeitliche und räumliche Präsentation und die Darstellung der Daten übernimmt somit das Anwendungsprogramm. Auch Funktionen für die Benutzerinteraktion können in diesem Fall durch das Anwendungsprogramm bereitgestellt werden. Aufgabe des MMDBMS ist es, Funktionen für die Medientransformation und Medientranslation bereitzustellen und die geforderten Qualitätsparameter (QoS) zu garantieren. Für die Spezifikation dieser Funktionen und Parameter muß die Anfragesprache geeignete Konstrukte anbieten.

Bei der Anfrageformulierung über eine Benutzerschnittstelle müssen zusätzlich zur Medientransformation und Medientranslation auch Darstellungsfunktionen für die verschiedenen Medientypen durch das MMDBMS angeboten werden. Ebenfalls werden Funktionen für die Steuerung der Darstellung dieser Mediendaten und zur Unterstützung wesentlicher Benutzerinteraktionen benötigt, um zum Beispiel das Abspielen von Videosequenzen zu steuern. Eine Multimedia-Anfragesprache muß somit Konstrukte für die Medientranslation/-formation, für die Beschreibung der QoS-Parameter und für die Festlegung der Darstellung der Medienobjekte anbieten.

In letzter Zeit zeichnet sich eine Entwicklung im Bereich der Multimedia-Datenbanken ab, immer mehr Präsentationsfunktionalität in das MMDBMS zu verlagern. Einige Ansätze [LSB⁺99] gehen sogar so weit, komplette Präsentationen als eine Art Sicht abzuliegen und die gesamte Präsentationsfunktionalität, wie räumliche Anordnung, Synchronisation usw., durch das MMDBMS bereitzustellen. Hierfür muß eine Anfragesprache sämtliche Präsentationsparameter zusätzlich zur eigentlichen Anfrage spezifizieren.

4.3 Zusammenfassung

Dieser Abschnitt faßt die aufgestellten Anforderungen an Multimedia-Anfragesprachen zusammen. Im ersten Teil dieses Kapitels wurden allgemeine Anforderungen für eine Anfragesprache betrachtet, wobei die Gültigkeit der einzelnen Kriterien im Zusammenhang

Kriterium	Erklärung
Ad-hoc Formulierung	Formulierbarkeit von Anfragen, ohne ein vollständiges Anwendungsprogramm zu schreiben
Generische Operatoren	Anfragesprache besteht aus einer kleinen Menge generischer Operatoren
Anwendungsunabhängigkeit	Sprachentwurf unabhängig von bestimmten Anwendungen
Deskriptivität	Beschreibung der gesuchten Daten und der Art der Präsentation dieser Daten ohne Vorgabe einer Berechnungsvorschrift
Mengenorientiertheit	Unterstützung von Datenmengen und Mengenoperationen
Orthogonalität	beliebige Kombination von Operationen, solange eine Typübereinstimmung gegeben ist
Effizienz	effiziente Realisierbarkeit der Kernoperationen
Erweiterbarkeit	einfache Erweiterung der Anfragesprache bei Erweiterungen des Datenbankmodells
Abgeschlossenheit	Darstellbarkeit der Resultate von Anfragen und Methodenaufrufen im Datenbankmodell
Adäquatheit	Abdeckung aller Konstrukte des Datenmodells durch die Anfragesprache (räumliche, zeitliche Beziehungen usw.)
Vollständigkeit	Sprache besitzt mindestens die Mächtigkeit der Relationenalgebra
Sicherheit	Terminierung und endliche Ergebnisse für alle Anfragen
Optimierbarkeit	Abbildung der Sprache auf ein internes Modell zur Optimierung
Formale Semantik	formale Definition aller Sprachkonstrukte und Operationen
Eingeschränktheit	berechnungsunvollständige Anfragesprache

Tabelle 4.1: Allgemeine Anforderungen für Anfragesprachen

mit Anfragesprachen in MMDDBMS besonders untersucht wurde. Dabei kann festgestellt werden, daß alle allgemeinen Kriterien auch für Multimedia-Anfragesprachen ihre volle Gültigkeit behalten. In Tabelle 4.1 sind diese allgemeinen Kriterien noch einmal zusammen mit einer kurzen Beschreibung aufgelistet.

Der zweite Abschnitt dieses Kapitels befaßte sich mit den speziellen Anforderungen, die aufgrund der in Kapitel 3 vorgestellten Besonderheiten an eine Multimedia-Anfragesprache gestellt werden. Dazu wurden eine Reihe von Kriterien aufgestellt, welche noch einmal in Tabelle 4.2 in der ersten Spalte aufgeführt sind. Die zweite Spalte der Tabelle faßt die aus den Kriterien resultierenden Anforderungen an eine Multimedia-Anfragesprache in verkürzter Form zusammen. Die in diesem Kapitel aufgestellten Anforderungen werden im weiteren Verlauf dieser Arbeit als Grundlage für den Vergleich der später vorgestellten Multimedia-Anfragesprachen dienen.

Kriterium	Anforderung an die Multimedia-Anfragesprache
Universell	Unterstützung mehrerer Medientypen und medienübergreifender Anfragen
Inhaltsbasierte Anfragen	Möglichkeit der Formulierung von Anfragen an Featurewerte und die Struktur der Multimediadaten sowie eine Unterstützung iterativ verfeinernder Anfragen
Räumliche Anfragen	Verfügbarkeit räumlicher Prädikate und Operatoren für die Anfrageformulierung
Zeitliche Anfragen	Verfügbarkeit zeitlicher Prädikate und Operatoren für die Anfrageformulierung
Unschärfe Anfragen	Unterstützung von unscharfen Prädikaten, gewichteten Anfragetermen, Ähnlichkeitsoperatoren und Relevanzwerten
Präsentation	Verfügbarkeit von Prädikaten und Operationen für eine Medientransformation/-translation, Raum-/Zeit-Constraints und QoS-Parameter zur Präsentation der Anfrageergebnisse

Tabelle 4.2: Spezielle Anforderungen für Multimedia-Anfragesprachen

5 | Ausgewählte Ansätze für Multimedia-Anfragesprachen

Eine Vielzahl von Anfragesprachen für MMDBMS wurden in den vergangenen Jahren vorgeschlagen. Die Ziele, welche bei der Entwicklung der einzelnen Sprachen verfolgt wurden, sind dabei sehr unterschiedlich. Ein Teil der Sprachen entstand im Zusammenhang mit speziellen Anwendungen, wie zum Beispiel die Anfragesprache KEQL [CHIT98], deren Augenmerk auf medizinischen Anwendungen liegt. Andere Anfragesprachen sind auf bestimmte Medientypen spezialisiert oder unterstützen einzelne Anfragearten, wie zum Beispiel räumliche Anfragen.

Ziel dieses Kapitels ist es, die wichtigsten Anfragesprachen für MMDBMS anhand ihrer wesentlichen Merkmale vorzustellen. Dabei wurden hauptsächlich solche Anfragesprachen untersucht, die einen anwendungsunabhängigen Ansatz verfolgen, da dies eine Grundanforderung für Multimedia-Anfragesprachen darstellt. Die verschiedenen Ansätze sind dazu in drei Kategorien eingeteilt:

- neue, speziell entwickelte Anfragesprachen,
- Erweiterungen von SQL und
- Erweiterungen von OQL.

Diese drei Kategorien bestimmen im wesentlichen den Aufbau des Kapitels. Die Multimedia-Anfragesprachen, welche nun vorgestellt werden, sind dabei den einzelnen Kategorien zugeordnet. Die meisten vorgestellten Anfragesprachen sind Ansätze aus der aktuellen Forschung, welche teilweise noch nicht beendet ist. Dadurch existieren nur für die wenigsten Ansätze komplette Sprachbeschreibungen. Die nachfolgenden Beschreibungen stützen sich auf die in den Veröffentlichungen zu den einzelnen Anfragesprachen verfügbaren Informationen, welche unter Umständen nicht die volle Ausdrucksfähigkeit der Anfragesprachen beschreiben.

5.1 Neu entwickelte Anfragesprachen

In diesem Abschnitt werden vollständig neu entwickelte Anfragesprachen vorgestellt. Diese basieren nicht auf einer Standardsprache, wie SQL oder OQL. Ein Problem von neu entwickelten Sprachen ist die fehlende Akzeptanz beim Nutzer. Im allgemeinen ist

es schwierig, dem Benutzer zu verdeutlichen, warum er für jede Anwendung eine neue Sprache erlernen muß. Neuentwicklungen haben aber den Vorteil, daß man sich beim Entwurf der Sprache vollkommen auf die zu lösenden Probleme konzentrieren kann und nicht von den einer Standardsprache zugrunde liegenden Regeln abhängig ist. So lassen sich im Zusammenhang mit Multimedia-Anfragesprachen Konzepte für verfeinernde oder visuelle Anfragen in einer Neuentwicklung einfacher und komfortabler umsetzen.

5.1.1 PICQUERY+

PICQUERY+ [CIT⁺93, Pra97b] ist eine wissensbasierte, objektorientierte Anfragesprache, welche für Bild- und alphanumerische Datenbanken entworfen wurde. Ziel der Entwicklung war es, eine sehr umgangssprachliche und leicht benutzbare Anfragesprache für MMDBMS zu definieren, die auch grundlegende Anforderungen aus dem Bereich der medizinischen Bildverarbeitung unterstützt. Trotz der engen Verbindung zu medizinischen Anwendungen wurde auf eine anwendungsunabhängige Sprachdefinition geachtet.

Das Datenmodell

Das Datenmodell, welches PICQUERY+ zugrunde liegt, ist ein erweitertes objektorientiertes Modell. Für eine Reihe von häufig benötigten Objekten sind die Grundstruktur und einige Operationen bereits vordefiniert. So enthält ein Bildobjekt bereits bildbezogene Attribute, wie Farbverteilung, Gestalt und Ähnliches. Die Beziehungen zwischen den Objekten lassen sich durch Kardinalitäten (1:1, 1:N, M:N), durch Vererbungsbeziehungen oder zusätzlich durch zeitlich und evolutionäre Beziehungen modellieren. PICQUERY+ erweitert dazu als einzige der in dieser Arbeit untersuchten Anfragesprachen das Objektmodell um ein zeitlich-evolutionäres Datenmodell (TEDM), wodurch evolutionäre Merkmale beschrieben werden können, welche man unter anderem in medizinischen Anwendungen benötigt. So kann zum Beispiel die Entwicklung von bestimmten körperlichen Merkmalen im Laufe der Zeit abgebildet werden. Dazu werden folgende evolutionäre Konstrukte angeboten:

- *Evolution*: sich im Laufe der Zeit ändernde Eigenschaften von Objekten,
- *Fusion*: zwei oder mehr Objekte verschmelzen zu einem neuen Objekt und
- *Spaltung*: ein Objekt wird in zwei oder mehrere Objekte aufgespalten.

Für die Formulierung von Prädikaten in Anfragen werden verschiedene Operatoren unterstützt. Diese sind im einzelnen:

- arithmetische Vergleichsoperatoren: =, <, >, ≤, ≥ und ≠,

- zeitbezogene Operatoren für zeitliche Prädikate: *after, before, during, between, in, overlaps, meets, equivalent, adjacent, follows* und *precedes*,
- räumliche Operatoren zur Überprüfung der räumlichen Merkmale der Daten: *intersects, contains, is collinear with, infiltrates, left of, right of, above, below, in front of* und *behind*,
- evolutionäre Operatoren zur Beschreibung der verschiedenen Entwicklungsphasen eines Objektes: *involves into, fuses into, splits into* und
- für die Beschreibung unscharfer Anfragen der Operator: *similar to*.

Mit dem Operator für unscharfe Anfragen „similar to“ sind weitere Parameter verknüpft, welche ein Distanzmaß für die Ähnlichkeit angeben. Die genauen Parameter hängen dabei von den in der Wissensbasis dargestellten Beziehungen ab, da ein Ähnlichkeitsmaß vom speziellen Anwendungsgebiet abhängig ist.

Für die Beschreibung der Präsentation von Anfrageergebnissen enthält die Sprache mehrere Konstrukte, sowohl für den Aufruf von Bearbeitungsmethoden auf den Ergebnisobjekten als auch zur Beschreibung der Art der Darstellung. Die Bearbeitungsmethoden können, ähnlich wie Aggregatfunktionen, bestimmte Eigenschaften aus dem Anfrageergebnis ableiten. PICQUERY+ unterstützt hierbei Funktionen für numerische Daten (Statistiken usw.) und Funktionen für Bilddaten, wie Einstellung von Kontrast, Helligkeit und Größe, Graustufendarstellung, Farbhistogramme und verschiedene Filterfunktionen. Für die Art der Präsentation bietet PICQUERY+ ebenfalls unterschiedliche Methoden für numerische Daten und Bilddaten an. Numerische Daten können als Tabelle oder in verschiedenen Diagrammtypen präsentiert werden. Die Darstellung von Bilddaten kann beispielsweise als Diashow mit einem festen zeitlichen Abstand zwischen einzelnen Bildern erfolgen. Für eine ausführliche Beschreibung der verschiedenen Bearbeitungsmethoden und Präsentationsarten sei auf die Literatur [CIT⁺93] verwiesen.

Anfragen in PICQUERY+

Eine Anfrage in PICQUERY+ wird in drei Schritten ausgeführt. Als erstes wählt der Benutzer aus verschiedenen Datenbanken die gewünschte Datenquelle aus. Im zweiten Schritt werden die Bedingungen zur Auswahl der Daten spezifiziert. Ähnlich wie die Sprache QBE [HS97] gibt PICQUERY+ hierfür ein Tabellengerüst vor. Tabelle 5.1 zeigt ein solches Tabellengerüst, wobei die folgende Beispielanfrage in PICQUERY+ dargestellt ist: „Finde alle Bilder, auf denen ein rotes Auto links von einem blauen Auto steht.“

Zur Spezifikation der gesuchten Daten enthält die Tabelle fünf Spalten:

1. *Objekt-Definition*: Festlegung des Objektes oder eines Objektattributes, welches von Interesse ist,

Object	RO	Object Value	LO	Group
AUTO	IN	IMG.feature.object	AND	1
AUTO2	IN	IMG.feature.object	AND	1
AUTO.farbe	=	"blau"	AND	1
AUTO2.farbe	=	rot"	AND	1
IMG.feature.left	IS	(AUTO2,AUTO)		

Tabelle 5.1: Beispielanfrage in PICQUERY+

2. *Beziehungs-Operator (RO)*: Bestimmung des Operators für die Zuordnung der Objekte zu Objektwerten aus Spalte drei,
3. *Objektwert-Definition*: Angabe eines Objektwertes, der im Zusammenhang mit der Objektdefinition (Spalte eins) und dem Operator (Spalte zwei) ein Prädikat zur Einschränkung der Ergebnismenge bildet oder die Angabe eines Beispielobjektes für eine Ähnlichkeitssuche.
4. *Logischer Operator (LO)*: Definition des logischen Operators (*and*, *or*, *not*), mit welchem die aktuelle und die nachfolgende Zeile verknüpft werden, um komplexe Prädikate zusammensetzen.
5. *Gruppen-Definition*: Festlegung von Gruppierungen in komplexen Prädikaten (ähnlich der Klammerung von Termen in textbasierten Anfragen). Zeilen mit der gleichen Group-Nummer gehören dabei zu derselben Gruppe.

Nachdem die Eingabe der Prädikate in das Tabellengerüst abgeschlossen ist, werden durch das MMDBMS die passenden Objekte aus der zuvor angegebenen Datenbasis ausgewählt. Das Ergebnis der Anfrage wird ebenfalls in einem Tabellengerüst angezeigt. Anschließend folgt der letzte Schritt der Anfrageformulierung in PICQUERY+, die Festlegung der Ergebnispräsentation. Hierbei können durch den Benutzer, aus den im Tabellengerüst dargestellten Objekten, einzelne, interessante Objekte und Attribute für die Präsentation ausgewählt werden. Außerdem ist es möglich, weitere Berechnungen, wie zum Beispiel Bildbearbeitungsfunktionen, auf den ausgewählten Objekten auszuführen. Am Ende muß für die einzelnen Ergebnisobjekte eine der Darstellungsarten angegeben werden, die im Zusammenhang mit dem Datenmodell vorgestellt wurden.

Wie in den vorangegangenen Abschnitten zu sehen war, unterstützt PICQUERY+ einen dreistufigen Anfrageprozeß. Im ersten Schritt wird die Datenquelle spezifiziert, der zweite Schritt bestimmt durch die Angabe von Prädikaten die Ergebnismenge und der dritte Schritt entscheidet über die Präsentation der Ergebnisse einer Anfrage.

5.1.2 SCORE

SCORE [ATY+95] ist ein System für die inhaltsbasierte Ähnlichkeitssuche in Bilddaten und wurde an der Universität von Illinois/Chicago entwickelt. Zur Beschreibung

der Bildinhalte verwendet SCORE eine dem ER-Modell ähnliche Notation. Anfragen lassen sich über ein grafisches Benutzerinterface formulieren. SCORE unterstützt neben unscharfen Anfragen für Attributwerte und für nichträumliche Beziehungen auch eine Ableitung räumlicher Beziehungen, was nachfolgend noch genauer erläutert wird. Des Weiteren werden Relevanzwerte für die Anfrageergebnisse angeboten. Durch die Beschränkung auf Bilddaten bietet SCORE keine zeitlichen Prädikate und Operatoren. Eine einfache Präsentation der Anfrageergebnisse übernimmt das Benutzerinterface.

Das Datenmodell

SCORE ist ein Datenbanksystem für die Suche in Bilddaten. Das Datenmodell verwaltet hierfür ausschließlich Bild- und bildinhaltsbeschreibende Daten. Weitere Medientypen, wie Audio- und Videodaten, werden nicht angeboten. Zur Beschreibung des Bildinhaltes wird eine an ER-Modelle angelehnte Notation verwandt, die von den Autoren als „verfeinertes ER-Modell“ bezeichnet wird. Dieses Modell wird aber nicht wie üblich zur Schemadefinition verwendet, vielmehr wird für jedes einzelne Bild ein „verfeinertes ER-Modell“ definiert, welches den Inhalt des Bildes beschreibt. Wie die Konstrukte des ER-Modells zur Bildinhaltsbeschreibung zu verwenden sind, ist dabei fest vorgegeben:

- *Entities* repräsentieren die Objekte innerhalb eines Bildes (z.B. Personen oder Gebäude),
- *Attribute* charakterisieren die Eigenschaften der Objekte, wie Farbe, Größe, Position usw. und
- *Beziehungen* stellen die Zusammenhänge zwischen den Objekten dar, wobei nur die folgenden fest vorgegebenen Beziehungstypen möglich sind:
 1. *Handlungsbeziehungen*, diese Beziehungen beschreiben Handlungen, die zwischen den Entities auftreten (wie zum Beispiel: „Der Mann schüttelt die Hand der Frau.“) und
 2. *Räumliche Beziehungen* zur Abbildung der relativen räumlichen Position von Entities zueinander.

Sowohl die Handlungsbeziehungen als auch die räumlichen Beziehungen werden weiter unterteilt in symmetrische und gerichtete Beziehungen. Bei einer symmetrischen Beziehung \mathbb{R} zwischen zwei Entities A und B gilt $\mathbb{R}(A, B) = \mathbb{R}(B, A)$. Das obige Beispiel, „Der Mann schüttelt die Hand der Frau.“, ist eine solche symmetrische Beziehung, denn es gilt ebenso, „Die Frau schüttelt die Hand des Mannes.“ Eine gerichtete Beziehung \mathbb{R}' gilt nur für eine Richtung $\mathbb{R}'(A, B)$, es existiert aber eine korrespondierende inverse Beziehung. Die räumlichen Beziehungen, wie „links von“ oder „über“, sind Beispiele für gerichtete Beziehungen. Benötigt wird die Klassifikation in SCORE unter anderem für die Ableitung räumlicher Beziehungen aus bereits bestehenden. So kann zum Beispiel aufgrund der Transitivität räumlicher Beziehungen aus A oberhalb von B und B oberhalb von C

gefolgert werden, daß A oberhalb von C liegt. Hierfür stellt SCORE eine Menge von Ableitungsregeln bereit, die zur Schlußfolgerung von räumlichen Beziehungen und für die Eliminierung redundanter räumlicher Beziehungen genutzt werden.

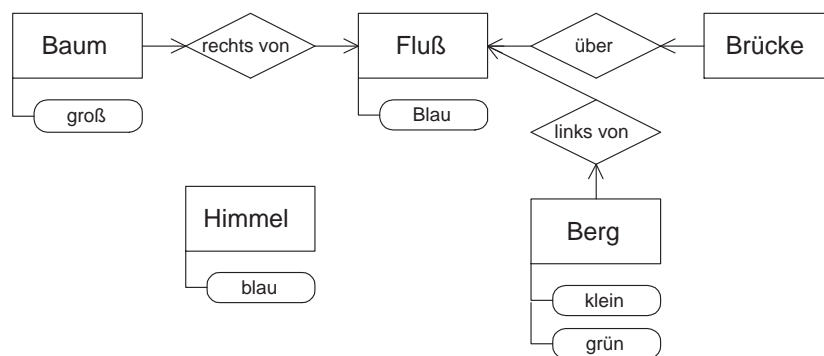


Abbildung 5.1: Beispiel für die Inhaltsbeschreibung von Bildern in SCORE

Das SCORE-System enthält somit eine Menge von „verfeinerten ER-Modellen“ zur Beschreibung der Bilddaten. Ein Beispiel für die Inhaltsbeschreibung eines Bildes durch ein solches „verfeinertes ER-Modell“ zeigt Abbildung 5.1. Der Begriff „verfeinertes ER-Modell“, ist in diesem Zusammenhang etwas irreführend, da im Bereich der Datenbank hierunter die Beschreibung des Datenmodells und nicht die Beschreibung der Daten selber wie im Falle von SCORE verstanden wird.

Anfragen in SCORE

SCORE bietet ein grafisches Benutzerinterface für die Formulierung von Anfragen und zur Eingabe der Beschreibung der Bilddaten an, welches in diesem Abschnitt beschrieben wird. Eine textbasierte Beschreibung von Anfragen ist nicht möglich. In SCORE wird eine Anfrage vielmehr durch eine Kombination von Beispielobjekten (dargestellt durch Entities) mit deren charakteristischen Merkmalen (Attribute) und Beziehungen ausgedrückt. Die fertig formulierte Anfrage stellt somit selbst wieder ein „verfeinertes ER-Modell“ dar. Dieses wird während der Anfragebearbeitung anhand spezieller Vergleichsmethoden mit den abgespeicherten Modellen verglichen, um so ähnliche Bilder zu bestimmen. Der Anfrageprozeß gestaltet sich dabei wie folgt:

1. Auswahl von Beispielobjekten: Hierfür steht eine Menge vordefinierter Beispiel-Entities zur Verfügung, die abhängig vom Anwendungsgebiet angepaßt werden können. Die passenden Entities werden aus Dialogboxen ausgewählt und in einer Anfragemaske angeordnet.
2. Verfeinern der Entities: Durch die Auswahl von relevanten Attributwerten werden die charakteristischen Merkmale der gesuchten Entities definiert. Dies kann sowohl durch grafische Methoden (z.B. Auswahl aus vordefinierten Listen) als auch durch Tastatureingaben geschehen.

3. Spezifikation der Beziehungen zwischen den Entities: Mit Hilfe grafischer Methoden werden räumliche Beziehungen und Handlungsbeziehungen zwischen den Beispiel-Entities angegeben, um so die Beziehungen, welche das gesuchte Bild enthalten soll, zu spezifizieren.
4. Anfragebearbeitung durch das SCORE-System: Durch die Schritte eins bis drei wurde stufenweise ein „verfeinertes ER-Modell“ aufgebaut. Das System führt nun einen Vergleich zwischen dem Anfrage-Modell und den abgespeicherten Modellen durch. Für diesen Vergleich sind geeignete Ähnlichkeitsmaße und Berechnungsalgorithmen definiert, für deren genaue Beschreibung auf [ATY⁺95] verwiesen sei.
5. Ergebnispräsentation: Die ähnlichsten Bilder werden dem Benutzer in Form einer Liste, bestehend aus einer verkleinerten Bilddarstellung und dem „verfeinerten ER-Modell“ der Bilder, präsentiert.
6. Verfeinerung der Anfrage: Mittels Browsen durch die Ergebnismenge kann der Benutzer hieraus zu seiner Anfrage passende Entities und Attributwerte auswählen und zu den bereits in der Anfragemaske vom vorherigen Anfrageprozeß enthaltenen Entities und Beziehungen hinzufügen. Anschließend beginnt der Anfrageprozeß wieder bei Schritt 1, das heißt, neue Entities können ausgewählt oder bereits ausgewählte können gelöscht werden, Attribute lassen sich hinzufügen oder ändern. Dieser Iterationsprozeß kann solange fortgesetzt werden, bis der Benutzer mit dem Ergebnis zufrieden ist.

SCORE unterstützt somit einen iterativen Anfrageprozeß mit Relevanzfeedback durch den Benutzer, welcher aber in einer etwas abgewandelten Form abläuft, als er vom Information Retrieval bekannt ist. Beim klassischen Relevanzfeedback-Verfahren wird im allgemeinen nur auf der Ergebnismenge verfeinert, wogegen SCORE das Relevanzfeedback für eine Verfeinerung der Anfragebedingungen nutzt und eine erneute Anfrage auf dem gesamten Datenbestand ausführt.

5.1.3 TVQL

Mit TVQL (Temporal Visual Query Language) wird von Hibino und Rundensteiner in [HR96] eine visuelle Anfragesprache für Video-Beschreibungsdaten vorgeschlagen, die in diesem Abschnitt genauer untersucht werden soll. Ziel der Entwicklung dieser Sprache war die Unterstützung der Analyse von Videodaten bezüglich ihrer räumlichen und zeitlichen Charakteristik sowie die Ableitung von zeitlichen Trends aus den Beschreibungsdaten (zum Beispiel Ereignis A tritt immer gleichzeitig mit Ereignis B auf). Nachfolgend wird zuerst das Modell zur Beschreibung der Videodaten vorgestellt. Dieses bietet die Grundlage für die Formulierung von zeitlichen Anfragen mit Hilfe der visuellen Anfragesprache TVQL, welche im zweiten Teil dieses Abschnittes betrachtet werden.

Annotation Modell

Zur Beschreibung des Raum-/Zeitverhaltens und anderer semantischer Informationen, die in den Videodaten enthalten sind, verwendet TVQL ein auf Annotationen basierendes Datenmodell. Die Grundkonstrukte dieses Modells sind:

- *Video*, gekennzeichnet durch Startzeit und Endzeit eines Videos und einem Offsetwert. Zusätzlich gibt es systemweit einen einheitlich Zeitstrahl, der mit der Frame Nummer Null beginnt und auf welchem die Videos entsprechend dem Offsetwert angeordnet sind. Für die Umrechnung von Start- und Endzeit in Framenummern und umgekehrt werden geeignete Methoden angeboten.
- *Videosegment*, welches einen zeitlichen Ausschnitt aus einem Video darstellt.
- *Video-Annotation*, die bestimmte, für den Benutzer interessante, Videoausschnitte beschreiben. Sie sind zeitlich verknüpft mit Videosegmenten und beziehen sich auf einen räumlichen Ausschnitt des Videos (zum Beispiel, linke obere Ecke).

Jedem Video in TVQL wird eine endliche Menge von Video-Annotationen zugeordnet. Hierbei ist zur Beschreibung der Videodaten für die Video-Annotationsobjekte eine feste Struktur vordefiniert. Diese wird in Abbildung 5.2 grafisch dargestellt.

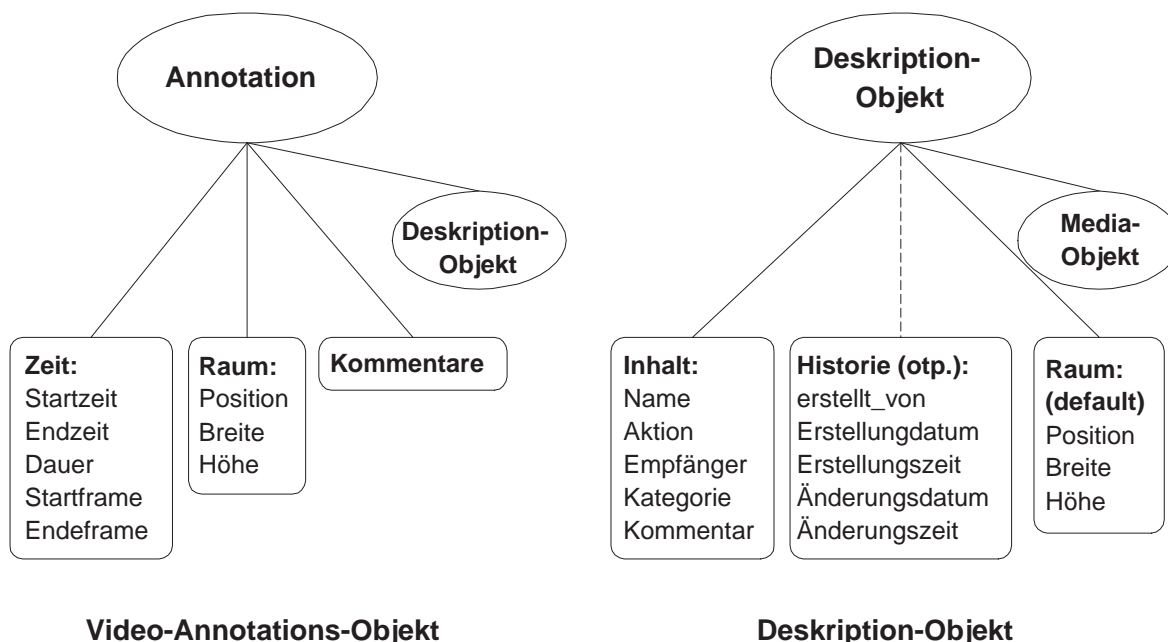


Abbildung 5.2: Objekte zur Beschreibung von Videodaten in TVQL

Eine Video-Annotation enthält vier Komponenten: zeitliche Informationen (Startzeit, Endzeit usw.), räumliche Informationen, freie Benutzerkommentare und eine Referenz

zu einem Deskription-Objekt. Dieses besteht wiederum aus vier Teilen, wie in Abbildung 5.2 zu sehen ist. Der erste Teil beschreibt semantische Informationen zu dem im Videoausschnitt dargestellten Ereignis, wie den Namen der Person oder des Objektes (Name) im Ausschnitt, die dargestellte Handlung, eine darin einbezogene weitere Person (Empfänger), die Zuordnung zu einer Kategorie und ein Kommentar zum Deskription-Objekt.

Der zweite Teil des Deskription-Objektes ist optional und gibt an, wann das Objekt erstellt oder zuletzt geändert wurde. Die räumlichen Daten bilden den dritten Teil des Deskription-Objektes und bieten zur Vereinfachung des Beschreibungsprozesses Standardwerte für die Video-Annotationsobjekte an. Der vierte Teil des Deskription-Objektes ordnet diesem ein Media-Objekt zu. Ein Media-Objekt ist ein Symbol, was in einer grafischen Darstellung der Videos an den zugeordneten Stellen angezeigt wird, um die Analyse zu erleichtern. Momentan müssen alle Videobeschreibungen von Hand durchgeführt werden, was sehr zeitaufwendig ist. In Zukunft sollen die Annotationen durch geeignete Verfahren zumindest teilweise automatisiert werden.

Visuelle Anfragen in TVQL

Nachdem das Beschreibungsmodell von TVQL vorgestellt wurde, soll nun die Formulierung von Anfragen an diese Beschreibungsdaten betrachtet werden. TVQL benutzt hierfür sogenannte „dynamische Anfragen“, wodurch sich der Anfrageprozeß stark von dem klassischer Datenbanken unterscheidet. Während über eine grafische Benutzeroberfläche die Auswahlkriterien spezifiziert werden, wird gleichzeitig das Anfrageergebnis zu der momentanen Auswahl angezeigt. Ändert man ein Auswahlkriterium, so ändert sich gleichzeitig das Anfrageergebnis. Grundlage für die Auswahl ist immer die gesamte Datenbasis, aus der durch die Auswahlkriterien die gewünschten Daten herausgefiltert werden.

Für die Spezifikation der Auswahlbedingungen stehen auf der Benutzeroberfläche eine Menge von Auswahllisten, entsprechend den Attributen des Video-Annotations-Objektes (Name, Handlung, Historie usw.), zur Verfügung. Eine Auswahlliste enthält dabei alle zum entsprechenden Attribut in der Datenbank vorhandenen Werte.

Neben den Auswahllisten zur Spezifikation der gesuchten Daten werden für zeitbezogene Anfragen Schieberegler verwendet, wie sie in Abbildung 5.3 dargestellt sind. Diese lassen sich in TVQL für verschiedene Arten zeitbezogener Anfragen einsetzen:

- zur Beschreibung der Zeitdauer des gesuchten Videosegmentes (z. B.: „Finde alle Videoausschnitte, wo der Lehrer länger als eine Minute spricht.“),
- zur zeitlichen Einschränkung des betrachteten Videoausschnittes (z. B.: „Finde alle Personen, die im Video zwischen der fünften und zehnten Minute zu sehen sind.“),
- zur Beschreibung von zeitlichen Beziehungen zwischen Zeitintervallen.

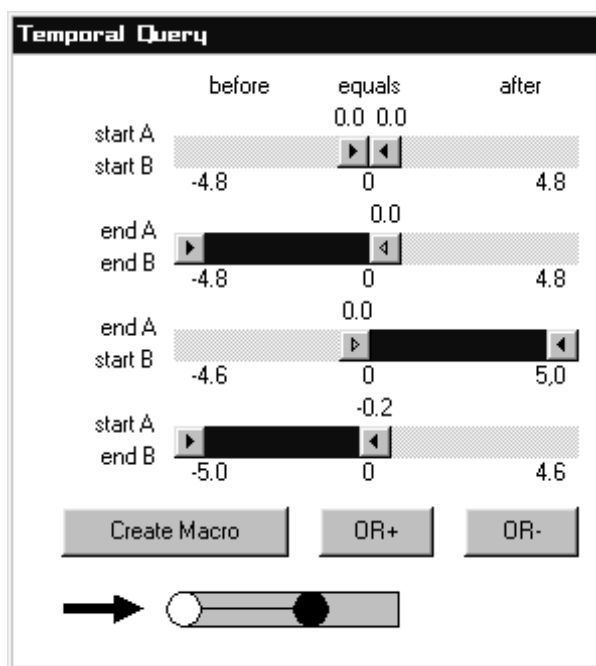


Abbildung 5.3: Beispiel einer visuellen zeitbezogenen Anfrage in TVQL

Anhand der zeitlichen Beziehungen soll die Verwendung der Schieberegler verdeutlicht werden. Bereits in Abschnitt 3.3.1 wurden 13 Zeitintervallbeziehungen vorgestellt, welche auch TVQL unterstützt. Jede dieser Beziehungen zwischen zwei Zeitintervallen A und B mit a_o (b_o) Startzeitpunkt von A (B) und a_e (b_e) Endzeitpunkt von A (B) kann durch die folgenden vier Differenzen-Constraints dargestellt werden:

$$a_o - b_o\theta, a_o - b_e\theta, a_e - b_o\theta, a_e - b_e\theta; \text{ wobei } \theta \in \{<, >, =\}.$$

Mit Hilfe eines Schiebereglers kann jeweils eines der obigen Differenzen-Constraints grafisch dargestellt werden. Hierfür enthält ein Schieberegler zwei bewegliche Komponenten. Mit diesen können die Relationen before ($<$), equals ($=$) und after ($>$) für die jeweils vor dem Schieberegler angegebenen Start-/Endzeitpunkte zweier Objekte eingestellt werden. Zusätzlich ist auch die Angabe von Zahlenwerten möglich. Unterhalb eines Reglers ist dazu der gesamte einstellbare Bereich angegeben, wohingegen der aktuell eingestellte Zahlenwert oberhalb der beweglichen Komponenten angezeigt wird. Im Beispiel in Abbildung 5.3 stehen bei dem obersten Schieberegler beide Komponenten auf dem Punkt Null. Dies entspricht dem Differenzen-Constraint $a_o - b_o = 0$ und bedeutet für zwei Zeitintervalle, daß diese gleichzeitig beginnen.

Verwendet man vier der Schieberegler, können somit die obigen Beziehungen zwischen Zeitintervallen grafisch spezifiziert werden. Zum Beispiel ist in Abbildung 5.3 die Beziehung A starts B dargestellt, wie sie in Tabelle 3.2 definiert wurde.

Der „Create Macro“-Button im unteren Teil des Beispiels dient zum Abspeichern von Reglereinstellungen für spätere Anfragen, wogegen der „OR+ Button“ vorgesehen

ist, um mehrere Intervallbeziehungen innerhalb einer Anfrage anzugeben. Dazu wird ein weiteres Fenster mit derselben SchiebereglerEinstellung angezeigt, die anschließend beliebig verändert werden kann. Die durch mehrere Fenster dargestellten Intervallbeziehungen werden bei der Anfragebearbeitung disjunktiv verknüpft. Mit Hilfe des „OR-Buttons“ läßt sich das entsprechende Fenster wieder entfernen. Das am unteren Ende des Beispiels hinter dem Pfeil dargestellte Symbol steht für die durch die aktuelle Reglereinstellung ausgewählte zeitliche Beziehung und wird bei Veränderung der Regler dynamisch angepaßt.

Durch die vorgestellte Anfragesprache TVQL lassen sich Anfragen an die Beschreibungsdaten und zeitbezogene Anfragen ausdrücken. Dabei bieten die grafischen Anfragemethoden und das Prinzip der dynamischen Anfragen, besonders zur Beschreibung zeitlicher Beziehungen, sehr gute Möglichkeiten. In einer anderen Arbeit [KR98] von Kaushik und Rundensteiner wird diese Technik ebenfalls für die Spezifikation räumlicher Anfragen benutzt. Leider bietet TVQL keine Unterstützung für weitere Medientypen außer Videodaten, und es besteht keine Möglichkeit, unscharfe Anfragen zu formulieren.

5.1.4 Framework für eine strukturierte Multimedia-Datenbank

Auf dem Gebiet der wissensbasierten Systeme, der Expertensysteme und der künstlichen Intelligenz sind Logikprogrammiersprachen sehr verbreitet. Die nachfolgend vorgestellte Multimedia-Anfragesprache basiert auf einer solchen formalen logischen Sprache zur Formulierung von Aussagen (Kalkül) und ist eine der wenigen Ansätze, die sich mit den mathematischen Grundlagen für Multimedia-Datenbanken befaßt.

Vorgestellt wird dieser Ansatz von S. Marcus und V. S. Subrahmanian in [MS96]. Sie definieren ein theoretisches Framework für eine strukturierte Multimedia-Datenbank, welches im nachfolgenden Unterabschnitt vorgestellt wird. Basierend auf diesem Framework wird daran anschließend eine logikbasierte Anfragesprache beschrieben. Im Laufe der Zeit wurden verschiedene Erweiterungen dieser Sprache und eine Abbildung des logikbasierten Sprachansatzes auf die Standardsprache SQL [Mar96] entwickelt, welche am Ende dieses Abschnittes erläutert wird.

Theoretisches Modell einer strukturierten Multimedia-Datenbank

Grundlage für das von Marcus und Subrahmanian vorgestellte Framework ist ein mathematisches Modell für eine Medieninstanz, welches als eine Verbindung zwischen der physischen und der logischen Repräsentation eines Medientyps betrachtet werden kann. Formal definiert ist eine Medieninstanz ein 8-Tupel $(\mathcal{S}, fe, ATTR, \lambda, \mathfrak{R}, \mathfrak{S}, Var_1, Var_2)$. Dabei ist:

- \mathcal{S} eine Menge von Medienobjekten, z.B. Bilder, Videosequenzen usw.,
- fe eine Featuremenge, womit entgegen der allgemeinen Definition von Features eine Menge von interessanten Objekten oder wichtigen Handlungen, enthalten in den

Medienobjekten, gemeint ist, z.B. Personen auf einem Foto, markante Szenen in einem Videofilm usw.,

- $ATTR$ eine Menge von Attributwerten, die den Medienobjekten und den Features zugeordnet werden,
- $\lambda : \mathcal{S} \rightarrow 2^{fe}$ eine Zuordnung von Mengen von Features zu Medienobjekten, um festzulegen, welches Feature in welchem Medienobjekt vorkommt,
- \mathfrak{R} eine Menge von Relationen auf $fe^i \times ATTR^j \times \mathcal{S}$ mit $i, j \geq 0$, die eine Zuordnung von Features, Attributwerten und Medienobjekten definiert. Zum Beispiel beschreibt die Relation `left_of(Baum, Haus, Foto1.gif)`, daß auf Foto1.gif ein Baum links von einem Haus zu sehen ist. In diesem konkreten Fall findet eine Zuordnung ($fe^2 \times ATTR^0 \times \mathcal{S}$) von zwei Features (Baum, Haus) zu einem Medienobjekt (Foto1.gif) statt.
- \mathfrak{S} eine Menge von Relationen auf \mathcal{S} , dadurch können Beziehungen zwischen den Medienobjekten definiert werden.
- Var_1 Menge von Objekten aus \mathcal{S} und
- Var_2 Menge von Objekten aus fe .

Mit Hilfe der obigen Definition einer Medieninstanz lassen sich nun verschiedene Medientypen formal beschreiben. Dies soll anhand eines kleinen Beispiels verdeutlicht werden. Die Bilddaten von sieben Fotos und den darauf abgebildeten Personen sind wie folgt als Medieninstanz beschreibbar:

1. Menge von Medienobjekten $\mathcal{S} = \{Bild1.gif, Bild2.gif, \dots, Bild7.gif\}$,
2. Features, zum Beispiel Personen auf den Fotos,

$$fe = \{Ingo, Stefan, Marco, Simone, Ingolf\},$$

3. Zuordnung, welche Personen auf welchem Bild zu sehen sind:

$$\lambda = \{(Bild1.gif, \{Ingo, Stefan\}), (Bild2.gif, \{Marco, Simone\}), \\ \dots (Bild7.gif, \{Ingo, Stefan, Marco, Ingolf\})\}$$

4. \mathfrak{R} könnte die räumliche Anordnung der Personen auf den Bildern beschreiben und
5. \mathfrak{S} könnte die Ähnlichkeit der Fotos bezüglich der Farbverteilung ausdrücken.

In ähnlicher Art und Weise kann man so auch Medieninstanzen für andere Medientypen definieren. Der Inhalt der Medienobjekte wird dabei durch die Features, Attributwerte

und die Relationen abgebildet. Wie im obigen Beispiel angedeutet, lassen sich auch Ähnlichkeitsbeziehungen ausdrücken. Eine Multimedia-Datenbank kann somit als eine endliche Menge von Medieninstanzen definiert werden. Diese Definition enthält zwei Schwachpunkte, sie kann nicht mit Synonymen in Features umgehen und sie bietet keine Möglichkeit zur Verschärfung oder Abschwächung von Anfrageprädikaten (vgl. Abschnitt 3.1.2). Beide Probleme werden durch die formale Definition einer strukturierten Multimedia-Datenbank (SMDS) behoben, die als ein 5-Tupel $(\{M_1, \dots, M_n\}, \equiv, \leq, inh, subst)$ dargestellt wird. Dabei ist:

- $M_i = (\mathcal{S}^i, fe^i, ATTR^i, \lambda^i, \mathfrak{R}^i, \mathfrak{S}^i, Var_1^i, Var_2^i)$ eine Medieninstanz,
- \equiv eine Äquivalenzrelation auf $\mathfrak{S} = \bigcup_{i=1}^n fe^i$ zur Definition von Synonymen (z.B. Auto \equiv Fahrzeug \equiv Wagen),
- \leq eine Halbordnung \mathfrak{S}/\equiv auf den Äquivalenzklassen von \mathfrak{S} , um Attributhierarchien, wie in Abschnitt 3.1.2 beschrieben, zu definieren,
- inh eine Abbildung $\mathfrak{S}/\equiv \rightarrow 2^{\mathfrak{S}/\equiv}$, die jedem Feature f eine Menge von Features zuordnet, welche bezüglich der Halbordnung (\leq) unterhalb von f liegen. Hierdurch wird eine Verfeinerung von Anfragetermen ermöglicht.
- $subst$ eine Abbildung von $\bigcup_{i=1}^n ATTR^i$ nach $2^{\bigcup_{i=1}^n ATTR^i}$, wodurch eine Zuordnung von Attributwerten zu anderen Attributwerten festgelegt wird, um dadurch eine Abschwächung von Anfragetermen zu erreichen.

Durch die in diesem Unterabschnitt vorgestellten Definitionen für eine Medieninstanz und eine strukturierte Multimedia-Datenbank ist eine formale mathematische Beschreibung für eine Multimedia-Datenbank gegeben. Darauf aufbauend haben Marcus und Subrahmanian eine logikbasierte Anfragesprache entwickelt, die nachfolgend beschrieben wird.

Logikbasierte Anfragen an ein SMDS

Um Anfragen an eine, durch das obige beschriebene Framework dargestellte, strukturierte Multimedia-Datenbank stellen zu können, wurde in [MS96] eine logikbasierte Sprache vorgestellt. Diese Sprache besteht aus folgenden Konstrukten:

1. *Konstanten:*

- (a) jedes $f \in fe^i, 1 \leq i \leq n$ ist eine Konstante (Featurekonstante) der Anfragesprache,
- (b) jedes $s \in \mathcal{S}^i, 1 \leq i \leq n$ ist eine Konstante (Medienobjekt-Konstante) der Anfragesprache,
- (c) jede ganze Zahl $i, 1 \leq i \leq n$ ist eine Konstante,

- (d) jedes Medium M_i ist eine Konstante (z.B. wenn $M_1 = video$, dann ist $video$ eine Konstante),
 - (e) eine endliche Menge von Attribut-Konstanten (Attributwerte, wie rot, blau, mittelgroß usw.).
2. *Funktionssymbol*: $flist$ ist eine Funktion, welche die Menge der Features zu einem Medienobjekt bestimmt,
 3. *Variablen*: unendliche Menge von Logikvariablen: V_1, \dots, V_i, \dots
 4. *Prädikate*:
 - (a) binäres Prädikat $frametype$, welches zu den zwei Argumenten (Medienobjekt, Medientyp) den Wert „wahr“ zurückliefert, genau dann, wenn der Medientyp des Medienobjektes mit dem im zweiten Parameter angegebenen Medientyp übereinstimmt.
 - (b) binäres Prädikat \in , überprüft das Vorkommen eines Features in einer Featuremenge,
 - (c) für jede j -stellige Beziehung $R \in \mathfrak{R}^i$ existiert ein j -stelliges Prädikat R^*
 - (d) für jede j -stellige Beziehung zwischen Feature und Medienobjekten $\psi \in \mathfrak{R}_2^i$ gibt es ein j -stelliges Prädikat ψ^*

Eine Term ist induktiv definiert:

1. Jede Konstante ist eine Term.
2. Jede Variable ist eine Term.
3. Wenn n ein n -stelliges Funktionssymbol und t_1, \dots, t_n Terme sind, dann ist auch $n(t_1, \dots, t_n)$ ein Term.

Spezielle Terme, die keine Variablen enthalten, sind Grundterme. Wenn p ein n -stelliges Prädikat ist und t_1, \dots, t_n Terme sind, dann ist $p(t_1, \dots, t_n)$ ein Atom. Eine Anfrage kann nun als Konjunktion von Atomen der Form:

$$(\exists)(A_1 \wedge \dots, A_n)$$

definiert werden. Anhand von zwei Beispielen soll die Verwendung dieser Sprache verdeutlicht werden. Bezogen auf das obige Beispiel einer Bilddatenbank mit sieben Bildern wird die Anfrage: „Finde alle Bilder, auf denen Marco zu sehen ist.“ wie folgt ausgedrückt:

$$(\exists s)(frametype(s, Bild) \wedge Marco \in flist(s)).$$

Der Buchstabe s ist hierbei ein Platzhalter für die Ergebnisobjekte. Durch das Prädikat $frametype$ wird der Typ des Medienobjektes angegeben und $flist(s)$ ist die Menge aller

Features, die dem Medienobjekt s zugeordnet sind. Da *Marco* im Beispiel als Feature (interessanter Ausschnitt eines Medienobjektes) definiert war, werden so alle Bilder, die *Marco* enthalten (Bild2.gif und Bild7.gif), als Ergebnis bestimmt. Im zweiten Beispiel werden alle Bilder gesucht, auf denen *Marco* links von *Simone* abgebildet ist. Die Anfrage hat folgende formale Darstellung:

$$(\exists s)(\text{frametype}(s, \text{Bild}) \wedge \text{Marco} \in \text{flist}(s) \wedge \text{Simone} \in \text{flist}(s) \\ \wedge \text{left_of}(\text{Marco}, \text{Simone}, s)).$$

In [MS96] wird von Marcus und Subrahmanian nachgewiesen, daß die vorliegende formale logikbasierte Anfragesprache adäquat bezüglich dem formalen Modell einer strukturierten Multimedia-Datenbank ist. Bei entsprechender Spezifikation von räumlichen, zeitlichen Beziehungen und einem Ähnlichkeitsmaß für eine Medieninstanz können somit durch die Sprache räumliche, zeitliche und Ähnlichkeitsanfragen formuliert werden. Ebenso wird eine Verschärfung/Abschwächung von Anfragetermen angeboten.

SQL-Anfragen an ein SMDS

Ein Problem des logikbasierten Ansatzes ist die komplizierte Formulierung von Anfragen, welche ein umfangreiches Wissen über Logikprogrammierung vom Benutzer verlangt. Deshalb werden diese Sprachen meist auf einer unteren Ebene eingesetzt, da sie eine gute mathematische Basis für den Anfrageprozeß und die Optimierung von Anfragen bieten. Eine Verwendung als Endbenutzersprache ist eher selten, meist werden für diesen Zweck die Konstrukte der logikbasierten Anfragesprache in eine andere Sprache (z.B. SQL) abgebildet oder durch eine grafische Benutzeroberfläche bereitgestellt.

Eine solche Abbildung auf die Standardsprache SQL wurde auch für die oben beschriebene logikbasierte Anfragesprache vorgenommen. In [Sub98] stellt V.S. Subrahmanian eine SQL-Erweiterung für die Anfrageformulierung an eine strukturierte Multimedia-Datenbank vor, welche nachfolgend auch mit SMDS-SQL bezeichnet wird. Diese Anfragesprache enthält neben allen Standard-SQL Statements zusätzlich folgende Konstrukte:

- **select** <Medien-Entity>, wobei ein Medien-Entity definiert ist als:
 1. $m: [i, j]$, Continuous-Medienobjekt, dargestellt durch die Menge aller Frames zwischen den Segmenten i und j ,
 2. m , Nichtcontinuous-Medienobjekt oder
 3. falls m ein Medien-Entity und a ein Attribut von m ist, dann ist $m.a$ auch ein Medien-Entity.
- **from** <Medientyp> <Quelle> <M>, hierbei werden nur die Medienobjekte betrachtet, welche zum angegebenen Medientyp und zur Datenquelle passen. M ist für diese Menge von Medienobjekten ein Platzhalter.

- **where** <Term> **in** <func_call>, dabei ist **Term** eine Variable oder ein Objekt mit demselben Datentyp wie das Ergebnis von **func_call**, und **func_call** ist eine der fünf Basisfunktionen:
 1. **FindType**(Medienobjekt) liefert zum vorgegebenen Medienobjekt den Medientyp zurück,
 2. **FindObjWithFeature**(Feature) bestimmt die Menge aller Medienobjekte mit dem angegebenen Feature,
 3. **FindObjWithFeatureandAttr**(Feature, Attribut, Attributwert) berechnet die Menge aller Medienobjekte, welche zum vorgegebenen Feature eine durch Attribut und Attributwert beschriebene Eigenschaft besitzen,
 4. **FindFeaturesinObj**(Medienobjekt) liefert alle im Medienobjekt enthaltenen Features (wesentliche Objekte),
 5. **FindFeaturesandAttrinObj**(Medienobjekt) berechnet für alle im Medienobjekt enthaltene Features eine Relation mit dem Schema (Feature, Attribut, Attributwert).

Mit Hilfe dieser SQL-Erweiterung kann nun die Beispielanfrage aus dem vorhergehenden Abschnitt: „Finde alle Bilder, auf denen Marco links von Simone abgebildet ist.“, in SMDS-SQL wie folgt dargestellt werden:

```

select  M
from    smds source1 M
where   FindType(M)=Bild and
        M in FindObjWithFeature (Marco) and
        M in FindObjWithFeature (Simone) and
        left_of (Marco, Simone, M)

```

Die in diesem Abschnitt vorgestellte Anfragesprache SMDS-SQL ermöglicht dieselben Anfragen, wie der zuvor beschriebene logikbasierte Sprachansatz. Ein Beweis hierfür wird in [Mar96] geführt.

5.2 Erweiterungen von SQL

Die meisten der vorgeschlagenen Ansätze für eine Multimedia-Anfragesprache basieren auf einer Erweiterung der relationalen Anfragesprache SQL, der ANSI-ISO-Norm für relationale Datenbanksprachen. Der Schritt der Erweiterung von SQL liegt nahe, denn viele MMDBMS wurden ebenfalls als Erweiterung relationaler DBMS entwickelt.

Die Verwendung einer Standardsprache, wie SQL, hat den Vorteil, daß sie durch ihre weite Verbreitung und internationale Standardisierung beim Benutzer im allgemeinen akzeptiert wird. Die theoretischen Grundlagen sind weitestgehend erforscht und eine formale Semantik der Grundkonstrukte und Operationen liegt bereits vor.

Es gibt aber auch Nachteile, welche sich durch die Verwendung von SQL als Basis ergeben. So ist die Grundstruktur einer Anfrage durch den SFW-Block fest vorgegeben. Auch der in Abschnitt 3.4 vorgestellte iterative Anfrageprozeß, welcher für unscharfe Anfragen von großer Bedeutung ist, läßt sich nur umständlich und benutzerunfreundlich umsetzen, da eine SQL-Anfrage im allgemeinen nach der Ergebnisdarstellung beendet ist. Um Anfrage-Attribute zu ändern oder um zusätzliche Attribute anzugeben, muß die komplette Anfrage neu formuliert werden. Eine Verfeinerung der erhaltenen Ergebnismenge kann nur durch das Anlegen einer temporären Relation erreicht werden. Dieser temporären Relation wird die Ergebnismenge zugewiesen, auf der dann verfeinernde Anfragen möglich sind. In den nachfolgenden Unterabschnitten werden einige SQL-erweiternde Sprachvorschläge vorgestellt.

5.2.1 KEQL

In diesem Abschnitt wird die Anfragesprache KEQL (knowledge-based spatial evolutionary query language) als eine SQL-basierte Sprache vorgestellt. KEQL [CHIT98] wurde, wie die zu Beginn dieses Kapitels betrachtete Anfragesprache PICQUERY+, an der Universität von Kalifornien (UCLA) entwickelt und unterstützt ebenfalls evolutionäre Merkmale von Multimedia-Daten, wie sie bereits bei PICQUERY+ beschrieben wurden. Ein direkter Zusammenhang zwischen den beiden Sprachen wird von den Autoren allerdings nicht hergestellt. Das Datenmodell und die Anfragekonstrukte von KEQL unterstützen ausschließlich Bilddaten und dazugehörige Beschreibungsdaten. Die Entwicklung von KEQL erfolgte im Zusammenhang mit der Implementierung einer medizinischen Bilddatenbank, basierend auf einem objektorientierten Datenbanksystem. Um einen Eindruck von den Möglichkeiten zu vermitteln, welche KEQL bietet, werden nachfolgend zuerst das zugrunde liegende Metadatenmodell und anschließend die Konstrukte der Anfragesprache vorgestellt.

Das Drei-Ebenen-Metadatenmodell

Der Inhalt der Bilddaten wird im KEQL-System durch Metadaten beschrieben. Hierzu wird ein Metadatenmodell vorgeschlagen, das aus drei aufeinander aufbauenden Ebenen besteht. Die Abbildung 5.4 stellt diese drei Ebenen grafisch dar.

Auf der Darstellungsebene werden die Eigenschaften der rohen Bilddaten, wie Farb-tiefe, Höhe und Breite eines Bildes, beschrieben und zusätzliche medizinische und technische Daten den Bildern zugeordnet. Letztere widersprechen dabei eindeutig dem Prinzip der Anwendungsunabhängigkeit. Des weiteren enthält die Darstellungsebene die Beschreibung von Bildausschnitten, die für den Benutzer interessante Objekte darstellen (z.B. Personen auf einem Foto). Für die Repräsentation dieser Bildausschnitte im Datenmodell wird eine Punktmengendarstellung (vgl. Abschnitt 3.2.1) verwendet, um eine möglichst hohe Genauigkeit der Beschreibung zu erreichen.

Die Semantikebene bildet räumliche und evolutionäre Beziehungen zwischen den auf der Darstellungsebene aus den Bildern extrahierten Objekten ab. Alle in Abschnitt 3.2

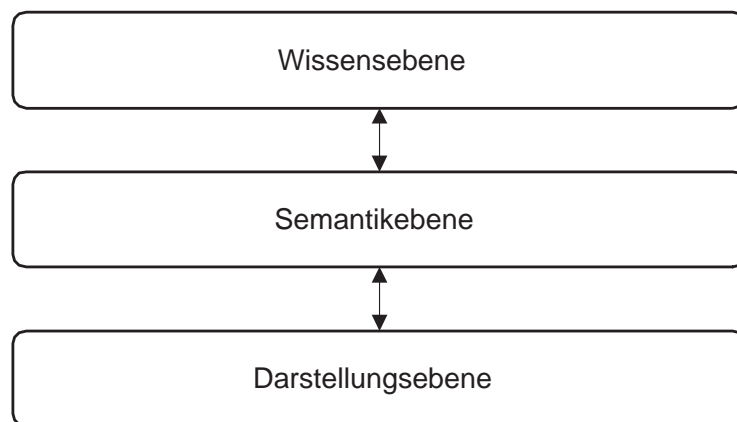


Abbildung 5.4: Drei-Ebenen des Metadatenmodells von KEQL

vorgestellten Arten von räumlichen Beziehungen (topologische, Richtungs- und Distanzbeziehungen) werden hierfür durch das Datenmodell unterstützt. Außerdem können die gleichen evolutionären Beziehungen (Spaltung, Fusion, Evolution) wie in der Sprache PICQUERY+ benutzt werden.

Basierend auf den in der Semantikebene dargestellten Objekten und Beziehungen werden auf der Wissensebene ähnliche Objekte in verschiedenen Begriffshierarchien (vgl. Abschnitt 3.1) zusammengefaßt. In KEQL nennt sich diese Hierarchie auch Typ-Abstraktions-Hierarchie (TAH), wobei für fast alle Attribute und Features eine solche TAH existiert. Benutzt werden diese TAH's unter anderem für Ähnlichkeitsanfragen und zur Unterstützung unscharfer Anfrageprädikate. Auch das Verschärfen bzw. Abschwächen von Anfragetermen ist mit Hilfe der TAH's möglich.

Anfragen in KEQL

Als SQL-basierte Anfragesprache benutzt KEQL selbstverständlich den bekannten „SFW“-Block, um Anfragen zu formulieren. Dieser wird um die zusätzliche Klausel `<Operations>` erweitert. Mit Hilfe dieser Klausel können Bildbearbeitungsfunktionen (Bildkonturen verstärken, Bild drehen, Zoomen) und Präsentationseigenschaften (Darstellungsposition und -ort, Diashow für Ergebnisbilder) für die Anfrageergebnisse festgelegt werden. Eine Anfrage in KEQL hat somit folgende allgemeine Form:

```

select    <select-list>
from     <from-list>
where    <condition>
           <operations>

```

Während die **select**-Klausel und die **from**-Klausel den gleichen Funktionsumfang wie die im Standard-SQL besitzen, wird die **where**-Klausel um zusätzliche Konstrukte für

die Formulierung von räumlichen, zeitlichen, evolutionären und unscharfen Anfragen erweitert. Folgende Prädikate und Operationen werden dabei unterstützt:

- Für *räumliche Anfragen* werden Prädikate zum Vergleich von topologischen, Richtungs- und Distanzbeziehungen angeboten. Für die Distanzbeziehungen stellt KEQL zusätzlich zu den Vergleichsoperatoren ($<$, \leq , $=$, \geq , $>$) einen „unscharfen Gleichheitsoperator“ ($\sim=$) bereit, welcher auch Ergebnisse zulässt, die in der Umgebung des angegebenen Wertes liegen.
- *Zeitliche Anfragen* beziehen sich in KEQL auf den Wert von Objekten zu bestimmten Zeitpunkten bzw. in bestimmten Zeitspannen. Hierfür werden Operatoren zur Abfrage von Start-, End-, Aufnahme- und Ereigniszeitpunkt sowie Vergleichsoperatoren auf Zeitintervallen (**before**, **after**, **during** usw.) angeboten.
- Speziell für medizinische Anwendungen unterstützt KEQL *evolutionäre Anfragen* durch die Prädikate **involved_from**, **fuses_into** und **splits_into**.
- Einen Ähnlichkeitsoperator (**similar_to**) und unscharfe Anfrageterme ermöglichen *unscharfe Anfragen* in KEQL. Der Ähnlichkeitsoperator benötigt ein zusätzliches Konstrukt (**based_on**), welches ein Feature oder ein Attribut angibt, bezüglich dessen die Operation ausgeführt wird. Unscharfe Anfrageterme sind zum Teil vordefiniert, wie **large** und **small** zur Größenbeschreibung räumlicher Objekte oder **far** und **near** für die Angabe von Distanzen zwischen Objekten. Abhängig vom Anwendungsgebiet können auch weitere unscharfe Anfrageterme vom Benutzer festgelegt werden.

Zur Verdeutlichung der obigen Prädikate und Funktionen und um einen Eindruck von der Anfragesprache KEQL zu vermitteln, ist nachfolgend die Anfrage: „Finde alle Bilder, die eine ähnliche Gestalt aufweisen, wie das auf dem Monitor angezeigte Bild. Zeige die gefundenen Bilder nacheinander auf dem Monitor.“ in KEQL dargestellt:

```

select    bild, bild.Image
from      Bilder bild, dummy
where     bild.Image similar_to dummy.Image
           (dummy.Image.selected_on_the_screen)
           based_on bild.Image.Gestalt
display sequences (bild.Image)

```

Der Bezeichner **dummy** in der obigen Anfrage dient als Platzhalter für das Bild auf dem Bildschirm, welches durch die spezielle Funktion **selected_on_the_screen** zugewiesen wird. Mit Hilfe der **display sequences**-Klausel erfolgt die Ausgabe der Ergebnisbilder.

Der Anfrageprozeß von KEQL unterstützt eine Verfeinerung der Anfrageergebnisse nach dem Berechnen der Anfrage. Mit Hilfe der Typ-Abstraktions-Hierarchien (TAH),

die im Zusammenhang mit dem Datenmodell erläutert wurden, lassen sich durch Auf- und Abwärtsverfolgung der Pfade dieser TAH's die Anfragebedingungen verschärfen oder abschwächen.

Für die Unterstützung eines solchen Anfrageprozesses und die Präsentation der Anfrageergebnisse sowie zur einfachen Formulierung von Anfragen wurde eine grafische Benutzeroberfläche entwickelt. Diese wird von den Autoren auch als visuelle Anfragesprache bezeichnet, da sämtliche Anfragen von KEQL grafisch ausgedrückt werden können. Unter dem Namen „MQuery“ geben Dionisio und Cardenas in [DC98] einen Überblick über diese visuelle Anfragesprache. Auf eine Vorstellung von MQuery wird in dieser Arbeit verzichtet, da sie denselben Funktionsumfang wie KEQL besitzt und auf demselben Datenmodell beruht.

Zusammenfassend kann festgestellt werden, daß die in diesem Abschnitt vorgestellte Anfragesprache KEQL gute Möglichkeiten für eine räumliche und unscharfe Suche in Bilddaten bietet. Leider beschränkt sich KEQL auf Bild und textbasierte Daten. Weitere Kritikpunkte sind die teilweise vorhandene Anwendungsabhängigkeit und die fehlende Orthogonalität der Anfragekonstrukte.

5.2.2 CSQL

Als eine weitere SQL-basierte Anfragesprache wird in diesem Abschnitt CSQL (Cognition and Semantic-based Query Language) [LC98] vorgestellt. Diese Sprache wurde für das Bilddatenbanksystem SEMCOG (SEMantics and COGNition-based image retrieval) entwickelt. Demzufolge unterstützt CSQL momentan nur Anfragen an Bilddaten und dazugehörige Metadaten, wobei die Systemarchitektur eine Erweiterung um zusätzliche Datentypen zuläßt. Ähnlich wie die Beschreibung der anderen Sprachansätze gliedert sich dieser Abschnitt in zwei Teile. Zuerst wird das Datenmodell von SEMCOG vorgestellt und anschließend erfolgt eine Beschreibung der Anfragesprache CSQL.

Das Datenmodell

Wie eingangs bereits erwähnt, werden ausschließlich Bilddaten in SEMCOG verwaltet. Ein Bild wird dabei als ein zusammengesetztes Objekt betrachtet, das mehrere Komponenten enthält. Jede dieser Komponenten entspricht einem interessanten Bildausschnitt. Auch eine Komponente kann wiederum weitere Komponenten enthalten. Die Bildstruktur wird in SEMCOG als eine Hierarchie dargestellt, wobei das ganze Bild an der obersten Stelle dieser Hierarchie steht und die Komponenten entsprechend darunter zugeordnet sind.

Bei der Beschreibung des Bildinhaltes wird zwischen den semantischen Merkmalen und den visuellen Merkmalen des Bildes unterschieden. Visuelle Merkmale¹ sind direkt aus dem Bild ableitbar, wie zum Beispiel Farbverteilung oder Gestalt. Semantische

¹in anderen Sprachen auch als Features bezeichnet

Merkmale (z. B. das Bild zeigt ein Auto) beschreiben die reale Weltbedeutung der dargestellten Objekte und können im allgemeinen nicht automatisch erkannt werden.

Entsprechend der Einteilung nach visuellen und semantischen Bildbeschreibungsdaten werden für die Darstellung im Modell Semantik-Entities und Bild-Entities unterschieden. Zusätzlich gibt es noch eine Mischform dieser beiden Entitytypen, die Dual-Entities. Beziehungen zwischen diesen Entitytypen können Contain-Beziehungen oder räumliche Beziehungen sein. Eine Contain-Beziehung besteht zum Beispiel zwischen einem Bildausschnitt und einem Bild, wenn dieser Ausschnitt in dem Bild enthalten ist. Räumliche Beziehungen kennzeichnen die räumliche Lage zweier Bildkomponenten zueinander. In SEMCOG werden hierfür nur die vier Richtungsbeziehungen (*links von*, *rechts von*, *oberhalb von* und *unterhalb von*) angeboten.

Mit Hilfe der vorgestellten Entitytypen wird in SEMCOG der Inhalt der Bilder und Bildkomponenten beschrieben. Die Contain-Beziehungen und die räumlichen Beziehungen verknüpfen diese zu den bereits erwähnten Bildhierarchien. Die SEMCOG-Datenbank enthält somit eine Menge von Bildhierarchien, die den Inhalt der einzelnen Bilder beschreiben. Außerdem wird für die Verschärfung und Abschwächung von Anfragetermen eine Begriffshierarchie angeboten. Über die genaue Struktur dieser Begriffshierarchie wird von den Autoren leider keine Aussage getroffen.

Anfragen in CSQL

CSQL erweitert die Standardsprache SQL durch zusätzliche Prädikate und Funktionen, um Anfragen an die Bilddaten formulieren zu können. Eine Anfrage in CSQL hat damit dieselbe „Select-From-Where“-Struktur, wie sie von SQL bekannt ist. Die von CSQL eingeführten Erweiterungen können anhand der verwendeten Entitytypen und Operatoren wie folgt eingeteilt werden:

- *Semantikbasierte Selektionskriterien:* Der Benutzer kann Bedingungen bezüglich der semantischen Merkmale der Bilder in verschiedenen Genauigkeitsstufen angeben. Folgende Prädikate sind dabei möglich:
 - *is:* Das Prädikat ist erfüllt, wenn beide Argumente identisch sind.
 - *is_a:* Das Prädikat ist erfüllt, wenn das zweite Argument eine Verallgemeinerung des ersten ist (z. B. Trabant *is_a* Auto).
 - *s_like:* Das Prädikat ist erfüllt, wenn beide Argumente einander ähnlich sind (z. B. Frau *s_like* Mann).

Argumente der obigen Prädikate können alle drei Arten von Entities, Semantik-, Bild- oder Dual-Entities, sein. Die Entscheidung, ob eine Verallgemeinerung bzw. Ähnlichkeit vorliegt oder nicht, wird mit Hilfe der Begriffshierarchie getroffen.

- *Bildbasiertes Selektionskriterium:* Für einen Ähnlichkeitsvergleich anhand visueller Merkmale zwischen Bild- und Dual-Entities wird das *i_like*-Prädikat eingeführt.

- *Räumliche Selektionskriterien:* CSQL unterstützt die räumlichen Prädikate *to_the_right_of in*, *to_the_left_of in*, *above_of in*, *below_of in*. Dabei werden zwei Entities (Semantik- oder Dual-Entity) und als drittes Argument ein Bild-Entity bezüglich der räumlichen Bedingung in dem angegebenen Bild verglichen und je nachdem, ob die Bedingung erfüllt ist, als Ergebnis der Wert „wahr“ oder „falsch“ zurückgeliefert.
- *Strukturbasiertes Selektionskriterium:* Das *contain*-Prädikat benötigt als erstes Argument ein Bild-Entity und als zweites ein Dual-Entity. Das Prädikat liefert den Wert „wahr“ zurück, falls das Dual-Entity im Bild-Entity als Komponente enthalten ist.

Prädikat	Argument 1	Argument 2	Argument 3
<i>is</i>	s/d/b	s/d/b	
<i>is_a</i>	s/d/b	s/d/b	
<i>s_like</i>	s/d/b	s/d/b	
<i>i_like</i>	d/b	d/b	
<i>is</i>	b	d	
<i>to_the_right_of in</i>	s/d	s/d	b
<i>to_the_left_of in</i>	s/d	s/d	b
<i>above_of in</i>	s/d	s/d	b
<i>below_of in</i>	s/d	s/d	b

Tabelle 5.2: Anfrageprädikate in CSQL

In Tabelle 5.2 sind alle vorgestellten Prädikate mit den entsprechenden Argumenten noch einmal zusammengefasst. Dabei stehen die Buchstaben s/d/b in den Argumentenspalten der Tabelle für Semantik-/Dual-/Bild-Entity. In SEMCOG wird die Anfragesprache CSQL lediglich für die interne Verarbeitung der Anfrage verwendet. Der Anwender formuliert seine Anfragen über eine grafische Benutzerschnittstelle. Diese stellt die Anfrageergebnisse dar, bietet die Möglichkeit der Anfrageverfeinerung und unterstützt die Spezifikation der Anfrage durch grafische Methoden. So können unter anderem Skizzen als Vorlagen für eine Ähnlichkeitssuche in den Bilddaten mit Hilfe der Benutzeroberfläche vorgegeben werden.

5.2.3 Medienspezifische abstrakte Datentypen und SQL/MM

Im folgenden soll beschrieben werden, wie sich die Standardanfragesprache SQL unter Ausnutzung der Erweiterungsmöglichkeiten von heutigen ORDBMS um zusätzliche Datentypen als eine Multimedia-Anfragesprache verwenden lässt.

Viele der heutigen, kommerziell verfügbaren ORDBMS bieten die Möglichkeit, über eine Schnittstelle zusätzliche Module einzubinden. Diese Module bestehen im wesentlichen aus einer Reihe benutzerdefinierter Datentypen und Funktionen sowie einem Schema (Tabellen, Assoziationen usw.), um beispielsweise Metadaten abzubilden. Dies kann unter anderem dazu genutzt werden, um Multimedia-Datentypen in ein ORDBMS zu integrieren. In der Literatur [Mar99] werden solche Module, die ein ORDBMS um zusätzliche Medientypen erweitern, auch medienspezifische abstrakte Datentypen (MADT) genannt.

Grundlage für die Erweiterungsmöglichkeiten von ORDBMS ist die SQL3-Norm. Diese bietet, wie bereits in Abschnitt 2.4 vorgestellt, unter anderem die Möglichkeit, abstrakte Datentypen (ADT) zu definieren, welche Attribute und Funktionen mit ihren Sichtbarkeitsstufen beschreiben. Darauf aufbauend beschäftigt sich ein Standardisierungsprojekt der ISO/IEC mit der Entwicklung von Standard-SQL-Klassenbibliotheken für Multimedia-Anwendungen. Diese Klassenbibliotheken stellen ebenfalls medienspezifische abstrakte Datentypen (MADT) dar.

Durch die MADT's wird das objektrelationale Modell um Multimedia-Datentypen und Funktionen erweitert. Die Anfragesprache SQL erhält hierdurch multimediaspezifische Prädikate und Funktionen und kann somit als Multimedia-Anfragesprache betrachtet werden. Die Möglichkeiten, welche diese Spracherweiterungen und die MADT's zur Formulierung von Multimedia-Anfragen bieten, sollen in diesem Abschnitt untersucht werden. Dazu sind nachfolgend die verschiedenen Klassenbibliotheken des SQL/MM-Standards dargestellt. Daran anschließend folgt eine kurze Vorstellung der von Industrie und Forschung bereits entwickelten MADT's. Die Einbindung in die Standardanfragesprache SQL und deren Möglichkeiten werden am Ende des Abschnittes betrachtet.

SQL Multimedia

Basierend auf der SQL3-Norm befaßt sich die ISO/IEC mit der Entwicklung von Standard-SQL-Klassenbibliotheken für Multimedia-Anwendungen. Unter dem Namen SQL Multimedia (SQL/MM) werden verschiedene Pakete für Volltextsuche und Textverarbeitung sowie für die Verwaltung von Multimediaobjekten, wie Bild, Audio, Animation und Video, standardisiert. SQL/MM ist ein mehrteiliger, sehr umfangreicher Standard, von dem einige Teile bereits abgeschlossen sind, wogegen sich andere noch in der Entwicklung befinden. Momentan besteht der SQL/MM-Standard [ISO95a, ISO95b, ISO97, ISO95c, ISO96] aus folgenden Teilen:

1. *Framework*: Dieses stellt den Standard als Ganzes vor, definiert die zu verwendenden Notationen und legt allgemeine Konzepte für die Klassenbibliotheken fest. Außerdem beschreibt das Framework, wie mit Hilfe des ADT-Konzeptes der SQL3-Norm abstrakte Datentypen zu definieren sind.
2. *Full-Text* definiert Datentypen und Methoden für die Volltextsuche und Textverarbeitung.

3. *Spatial* definiert Datentypen und Methoden für die Suche und Verwaltung räumlicher Daten.
4. *General Purpose Facilities* definiert eine Reihe allgemein benötigter Datentypen und Methoden für verschiedene Anwendungsgebiete, wie zum Beispiel Vektoren, komplexe Zahlen, trigonometrische und Exponentialfunktionen.
5. *Still Image* definiert Datentypen und Methoden für die Verwaltung von Bilddaten.

Für die Volltextsuche werden im wesentlichen ein Datentyp **FullText** mit Funktionen für die Konvertierung von bzw. in einen String und ein **contain**-Prädikat definiert, welches testet, ob eine Textphrase in einem **FullText**-Objekt enthalten ist. Des Weiteren werden einige Datentypen für die Verwaltung und die Suche in einem strukturierten Text vorgegeben.

Datentypen für die Abbildung von 2- und 3-dimensionalen Koordinaten und geometrischen Objekten werden im Teil 3 von SQL/MM [ISO97] definiert. Ein geometrisches Objekt ist dabei eine beliebige Aggregation aus Punkten, Linien und Flächen. Weiterhin bietet diese SQL-Klassenbibliothek räumliche Funktionen wie:

- **Buffer** berechnet zu einem geometrischen Objekt und einem Abstand n eine Fläche, für die gilt: Jeder Punkt der Begrenzung dieser Fläche hat den gleichen Abstand n zum geometrischen Objekt.
- **Centroid** bestimmt den Mittelpunkt eines geometrischen Objektes,
- **Envelop** liefert für ein räumliches Objekt als Funktionsergebnis das minimale umschreibende Rechteck dieses Objektes,
- **Area** berechnet die Fläche des räumlichen Objektes und
- **Perimeter** bestimmt den Umfang eines räumlichen Objektes.

Räumliche Prädikate werden ebenfalls unterstützt für den Test auf Berührung (meets), Überlappung (overlaps), Enthaltensein (contains, contained by) und die Disjunktion (outside) von zwei räumlichen Objekten.

Der Teil 4 des SQL/MM-Standards [ISO95c] definiert eine Klassenbibliothek mit häufig benötigten allgemeinen Datentypen und Funktionen, die im Prinzip nichts mit Multimedia zu tun haben. Im einzelnen sind dies: der Datentyp Winkel (Angel) mit verschiedenen Winkelfunktionen (Sinus, Kosinus, Tangens usw.), ein Datentyp für komplexe Zahlen mit den Grundrechenoperationen, ein Vektor- und ein Array-Datentyp.

Für die Verwaltung von Bilddaten legt Teil 5 des SQL/MM-Standards [ISO96] einen abstrakten Datentyp **StillImage** fest. Dessen Typdefinition enthält lediglich die Bilddatei als Binary Large Objekt und einige bildbeschreibende Attribute, wie Höhe, Breite, Farbtiefe, Auflösung und Zeitpunkt des Einfügens sowie ein Attribut für Kommentare

zum Bild. Weitere Funktionen für Bildobjekte oder die Ableitung von Features und darauf aufbauende Suchfunktionen sind nicht vorgesehen.

Neben den oben beschriebenen bereits größtenteils abgeschlossenen Teilen des SQL/MM-Standards sollen zukünftig weitere Klassenbibliotheken für Multimedia-Anwendungen standardisiert werden. Vorgesehen sind unter anderem abstrakte Datentypen für Audio- und Videodaten.

MADT's aus Industrie und Forschung

Eine weitere Möglichkeit für die Erweiterung des objektrelationalen Modells bieten, neben den zuvor beschriebenen Klassenbibliotheken des SQL/MM-Standards, die in Industrie und Forschung entwickelten medienspezifischen abstrakten Datentypen. Je nach Hersteller und Datenbanksystem heißen diese DataBlade, Extender oder Data Cartridge. Das zugrunde liegende Prinzip ist dabei in allen Fällen das gleiche, das Datenmodell wird um zusätzliche Multimedia-Datentypen und Methoden erweitert. Da ein umfangreicher Überblick über die MADT's der verschiedenen Hersteller den zeitlichen Rahmen dieser Arbeit sprengen würde, sollen nachfolgend nur einige wesentliche Erweiterungsmodule des Datenbanksystems Informix kurz vorgestellt werden, um so einen Eindruck von den Möglichkeiten zu vermitteln, welche die MADT's für die Formulierung von Anfragen an Multimedia-Daten bieten.

Excalibur Text Search DataBlade [Inf97b]: Dieses Erweiterungsmodul stellt Funktionen für die Volltextsuche bereit. Hierzu wird ein Text-Datentyp definiert. Die zugeordneten Textdokumente können sowohl in der Datenbank als auch im Filesystem abgelegt werden, wobei verschiedene proprietäre Datenformate, wie MSWord, HTML, PDF usw., unterstützt werden. Für die Suche nach Wortphrasen enthält das DataBlade eine **contain**-Funktion. Diese berechnet einen Ähnlichkeitswert zwischen der Wortphrase und dem Textobjekt und gibt den Ähnlichkeitswert als Ergebnis aus. Um eine effiziente Suche zu ermöglichen, werden durch das DataBlade entsprechende Indexstrukturen angelegt.

Excalibur Image DataBlade [Inf97a]: Durch dieses DataBlade werden ein Bild-datentyp und Funktionen für eine inhaltsbasierte Suche in Bildern anhand physischer Merkmale, wie Farbe, Textur und Gestalt, bereitgestellt. Ebenfalls definiert werden geeignete Indexstrukturen und Funktionen zur Feature-Extraktion, welche beim Einfügen eines Bildes in die Datenbank die physischen Merkmale ableiten. Ähnlich wie beim Text Search DataBlade kann eine Abspeicherung des Bildes in der Datenbank oder im Filesystem erfolgen. Für die angebotenen Datenformate GIF, JPEG, TIFF und BMP werden Konvertierungsfunktionen angeboten. Ähnlichkeitsanfragen können nur anhand eines Beispielbildes (Query by Example) formuliert werden, wobei für die Ergebnisbilder ein Relevanzwert angegeben wird. Des Weiteren enthält das Image DataBlade Funktionen, um bildbeschreibende Attribute, wie Höhe, Breite und Speicherformat eines Bildes, abzufragen.

Informix Video Foundation DataBlade: Von Informix wurde dieses DataBlade-

de als Ausgangsbasis für weitere, darauf aufbauende DataBlade's entwickelt. Bereitgestellt werden Datentypen und Methoden für Videodaten. Ebenfalls vordefiniert wird ein Metadatenschema für die Beschreibung der physischen Attribute (Länge, Format des Videostreams) und für eine Verwaltung von Szenenbeschreibungen.

GMD IPSI Continuous Long Field DataBlade [HSH⁺98]: Dieses DataBlade erweitert das Video Foundation DataBlade um Funktionen für die Echtzeitpräsentation von Continuous-Daten. Hierfür wird ein Continuous-Long-Field-Datentyp (CLF) definiert. Dies ist ein generischer Datentyp für beliebige Datenströme, zum Beispiel Audio- oder Videostreams. Voraussetzung für die Verwendung von CLF-Datentypen ist, daß die Daten auf einem ebenfalls bereitgestellten speziellen CLF-Server liegen.

GMD IPSI MPEG DataBlade [MHSA98]: Spezielle Datentypen und Methoden für Videodaten im MPEG-Format stellt dieses, ebenfalls auf dem Video Foundation DataBlade aufbauende, Erweiterungsmodul bereit. Momentan werden das MPEG1- und MPEG2-Format unterstützt. Für die Zukunft ist eine Erweiterung auf die Formate MPEG4 und MPEG7 geplant. Das DataBlade bietet Funktionen für den Zugriff auf den Datenstrom und die Dekodierung einzelner Frames. Außerdem werden eine Extraktion von Metadaten und eine Abspeicherung dieser im Metaschema des Informix Video Foundation DataBlade unterstützt.

SQL3 als Multimedia-Anfragesprache

Alle vorgestellten medienspezifischen abstrakten Datentypen basieren auf Konstrukten, die der SQL3-Standard zur Verfügung stellt. Dies gilt auch für die Formulierung von Anfragen, wobei die momentan auf dem Markt befindlichen Systeme nur einen Bruchteil der Konzepte von SQL3 umsetzen. Wie bereits in Abschnitt 2.4 erläutert, ist SQL3 abwärtskompatibel zum bekannten SQL-Anfrage-Konzept. Für die Anfrageformulierung steht somit der bekannte „SFW“-Block mit allen dazugehörigen Konstrukten zur Verfügung. Durch die zuvor vorgestellten medienspezifischen abstrakten Datentypen wird SQL3 um zusätzliche Datentypen, Funktionen und Prädikate erweitert. Speziell die Prädikate und Funktionen können für Anfragen an Multimedia-Daten benutzt werden. Als Beispiel ist nachfolgend die Anfrage: „Finde alle Dokumente, in denen der Wortlaut „Otto von Guericke“ vorkommt.“ in SQL3 dargestellt:

```
select *  
from mydocs d  
where contains (d.doc, "Otto von Guericke")
```

Die obige Anfrage benutzt das **contains**-Prädikat der FullText-Klassenbibliothek des SQL/MM-Standards. In der Datenbank wird eine Volltextsuche durchgeführt und alle Objekte, welche den Wortlaut enthalten, werden als Ergebnis zurückgeliefert. Bei der Verwendung des Informix Text Search DataBlade werden zusätzlich auch solche Textobjekte zurückgeliefert, welche den Wortlaut nur teilweise enthalten (z.B. nur das Wort

„Guericke“). Für die einzelnen Ergebnisobjekte wird in diesem Fall ein Relevanzwert angegeben.

Wie im obigen Beispiel bereits zu erkennen, hängen die Möglichkeiten für die Formulierung von Anfragen an Multimedia-Daten direkt von den angebotenen und in einem konkreten MMDBMS eingesetzten MADT's ab. Für den Vergleich der Fähigkeiten des MADT-Konzeptes in bezug auf Anfragen an Multimedia-Daten in Kapitel 6 wird deshalb von einem gleichzeitigen Einsatz aller vorgestellten MADT's in einem MMDBMS ausgegangen.

5.2.4 VideoSQL

VideoSQL ist die Anfragesprache eines objektorientierten Video-Informations-Datenbanksystems, genannt OVID [OT93]. In diesem System werden Videosequenzen und Beschreibungen von Video-Objekten, welche aus den Videosequenzen abgeleitet sind, verwaltet. Andere Medientypen, wie Audio- und Textdokumente, werden nicht unterstützt. Für die Eingabe und Änderung der Beschreibung einzelner Videosequenzen steht ein „Video-Object-Definition Tool“ zur Verfügung. Dieses unterstützt eine dynamische und inkrementelle Beschreibung, das heißt, die Inhaltsbeschreibung der Videoobjekte muß nicht mit einem Mal erfolgen, sondern kann im Laufe der Zeit verändert und verbessert werden. Die Suche in den Videodaten kann mit einem Browsing-Werkzeug „VideoChart“ oder durch Ad-hoc-Anfragen, formuliert in der Anfragesprache VideoSQL, erfolgen.

Das Datenmodell von OVID

Das Video-Object-Data-Model ist das dem OVID-System zugrunde liegende Datenmodell. Ein Videodokument wird dabei zerlegt in einzelne Videoframes (Einzelbilder). Ein Hauptbestandteil des Modells sind die Video-Objekte, welche den Inhalt eines bestimmten Videoausschnittes (Menge aufeinanderfolgender Videoframes) beschreiben. Ein Video-Objekt setzt sich zusammen aus:

- Objekt-Id,
- Zeitintervall, dargestellt durch Startframenummer und Endframenummer, in welchem das Video-Objekt bezüglich des gesamten Videofilms angeordnet ist,
- Menge von Attributen zur Inhaltsbeschreibung und
- grundlegende Methoden für die Videoverwaltung.

Durch die Objekt-Id wird jedem Video-Objekt ein eindeutiges Identifikationsmerkmal zugeordnet. Ein zeitlich festes Modell ordnet die Frames auf einer Zeitachse an, wodurch das Zeitintervall eines Videoausschnittes durch Start- und Endframenummer dargestellt werden kann. Für die Attribute, welche den Inhalt des Videoausschnittes beschreiben,

wird kein festes Datenbankschema vorgegeben. Der Nutzer kann und muß selber eine Attributstruktur festlegen oder andere Video-Objekte zur Inhaltsbeschreibung zuweisen. Zu den Grundfunktionen, die jedes Video-Objekt besitzt, gehören: Abspielfunktionen für Videoausschnitte (Abspielen, Pause, Stop, Vor, Zurück) und Funktionen für die Komposition und Dekomposition von Video-Objekten.

Die einzelnen Video-Objekte und Attribute stehen durch eine Spezialisierungs-/Generalisierungshierarchie (vgl. Abschnitt 3.1) miteinander in Beziehung. Dabei kann ein Video-Objekt auch Attribut eines anderen Video-Objektes sein. Des weiteren ist eine Vererbung von Attributen, basierend auf Interval-Inklusionsbeziehungen, möglich. Bei dieser im OVID-System definierten Beziehungsart kann ein Video-Objekt A Attribute und Attributwerte an ein anderes Video-Objekt B vererben, falls das Zeitintervall von Video-Objekt B im Zeitintervall von Video-Objekt A enthalten ist. Hierbei findet eine Vererbung zwischen Instanzen der Klasse Video-Objekt statt und nicht wie im klassischen objektorientierten Ansatz zwischen Klasse und Subklasse. Die Idee für diese Art der Vererbung beruht auf der Feststellung, daß häufig Videosequenzen andere Teilsequenzen enthalten, welche durch dieselben Attribute beschrieben werden können.

Für die Komposition von Video-Objekten enthält das Datenmodell zwei Operationen, *merge* und *overlap*. Die Merge-Operation generiert ein neues Video-Objekt C aus zwei existierenden Video-Objekten A und B . Das Video-Objekt C erbt dabei die Attribute der anderen beiden Objekte und bekommt als Zeitintervall die Vereinigung der Zeitintervalle der Video-Objekte A und B zugeordnet. Bei der Overlap-Operation wird ebenfalls ein neues Video-Objekt aus zwei anderen generiert. Das neue Video-Objekt enthält dabei den Durchschnitt der beiden anderen Objekte. Im einzelnen heißt dies, die geerbten Attribute sind die in beiden Ausgangsobjekten enthaltenen Attribute und das zugeordnete Zeitintervall ist der Durchschnitt der Zeitintervalle der Ausgangsobjekte. Diese Operation ist sinnvoll, um die Beschreibung von Szenen, welche durch mehrere Video-Objekte beschrieben werden, in einem Objekt zusammenzufassen.

Anfragen in VideoSQL

Als Anfragesprache des OVID-Systems unterstützt VideoSQL die Suche in Videodaten anhand ihrer Attributwerte. Das Ergebnis einer Anfrage ist eine Menge von Video-Objekten, welche die in der Anfrage spezifizierten Bedingungen erfüllen. Eine VideoSQL-Anfrage ist dabei wie folgt aufgebaut:

- SELECT Klausel: Diese Klausel unterscheidet sich stark vom Standard-SQL. Hier kann nur eine der Video-Objekt-Kategorien:
 - *Continuous*, ein einzelner zusammenhängender Videoausschnitt,
 - *Incontinuous*, mehrere einzelne Videoausschnitte und
 - *anyObject*, beide Arten *Continuous* und *Incontinuous*

für die Ergebnisobjekte vorgegeben werden. Weitere Konstrukte innerhalb der SELECT Klausel werden nicht unterstützt.

- FROM Klausel: Hier kann nur der Name einer Video-Datenbank angegeben werden.
- WHERE Klausel: Die WHERE Klausel gibt durch Attribute/Wertepaare und Vergleichsoperatoren die Prädikate für die Suche an. Folgende Prädikate stehen zur Verfügung:
 1. [Attribute] is [Value | Video-Objekt]: Hierbei werden Video-Objekte gesucht, die den spezifizierten Attributwert bzw. das Video-Objekt enthalten. Hat das Attribut in der Generalisierungshierarchie des zugrunde liegenden Datenmodells weitere untergeordnete Attributwerte, so werden diese ebenfalls für die Suche mitbenutzt.
 2. [Attribute] contains [Value | Video-Objekt]: Für Attribute, die als Mengentypen definiert sind, kann mit diesem Prädikat nach einzelnen Attributwerten in der Wertemenge der Attribute gesucht werden.
 3. defineOver [Videoausschnitt | Videoframe]: Ein und derselbe Videoausschnitt kann durch beliebig viele Video-Objekte beschrieben werden. Mit Hilfe dieses Prädikates werden alle Video-Objekte, die den angegebenen Videoausschnitt bzw. Videoframe beschreiben, ausgewählt.

Obwohl VideoSQL den bekannten „SFV“-BLOCK des Standard-SQL verwendet, muß man in diesem Zusammenhang eher von einer Einschränkung als von einer Erweiterung der Sprache SQL sprechen. Wie man anhand der oben aufgeführten Beschreibung der Sprachkonstrukte erkennen kann, unterstützt VideoSQL weder Aggregatfunktionen noch geschachtelte Anfragen. Die im Zusammenhang mit MMDBMS besonders wichtigen räumlichen und zeitlichen Beziehungen lassen sich gar nicht oder nur sehr umständlich über vom Anwender zugewiesene Attribute darstellen. Eine Formulierung unscharfer Anfragen ist nicht möglich. Für die Präsentation der durch die Anfrage ausgewählten Video-Objekte bietet das OVID-System eine Oberfläche mit den benötigten Abspiel-funktionen an. Über diese Oberfläche können auch die VideoSQL-Anfragen formuliert werden, wobei mehrere Anfrageergebnisse gespeichert und die darin enthaltenen Video-Objekte zu neuen zusammengesetzt (merge) werden können.

5.3 Erweiterungen von OQL

Wie bereits in Abschnitt 2.4 dargestellt, ist OQL [Cat94] eine objektorientierte, deklarative Anfragesprache, welche im Rahmen der Standardisierungsbemühungen der OMG vorgeschlagen wurde. Da in jüngster Zeit fast alle Ansätze für ein MMDBMS auf einem ODBMS aufbauen, existieren eine Reihe von Sprachvorschlägen für Multimedia-Anfragesprachen, die eine Erweiterung von OQL vorsehen. Einige dieser Sprachvorschläge sollen in den folgenden Unterabschnitten vorgestellt werden.

Für die auf OQL-Erweiterungen basierenden Anfragesprachen gelten im wesentlichen die gleichen Aussagen über Vor- und Nachteile, wie sie für SQL-basierte Sprachvorschläge

in Abschnitt 5.2 getroffen wurden. Positiv gegenüber dem SQL-basierten Ansatz ist die bessere Unterstützung iterativer Anfragen durch das Konzept der benannten Anfragen, welches OQL anbietet. Hierbei kann über den Namen einer Anfrage auf vorherige Anfrageergebnisse innerhalb einer anderen Anfrage zugegriffen werden, was eine Verfeinerung von Ergebnismengen erlaubt.

5.3.1 MMSQL

Die Multimedia Symbolic Query Language (MMSQL) wurde von Amato, Mainetto und Savino in [AMS98] im Zusammenhang mit einem Ansatz für ein System zur inhaltsbasierten Suche in Multimedia-Daten vorgestellt. Die Standardfunktionalitäten einer objektorientierten Anfragesprache werden hierbei um Konzepte für die Behandlung von unscharfen Anfragen erweitert. Für die Erläuterung des Sprachansatzes wird zunächst das Multimedia-Datenmodell vorgestellt. Daran anschließend werden die wichtigsten Konstrukte der Sprache erläutert.

Das Multimedia-Datenmodell von MMSQL

Als ein wichtiger Teil des von Amato et al. vorgestellten Ansatzes soll nun das objektorientierte Multimedia-Datenmodell beschrieben werden, da es die Grundlage für die darauf aufbauende Anfragesprache MMSQL bildet. Es setzt sich zusammen aus den drei nachfolgenden Teilmodellen:

- Multimedia Description Model (MDM), welches eine strukturierte Sicht auf die Mediendaten bietet,
- Multimedia Interpretation Model (MIM), für die Zuordnung einer semantischen Interpretation zu den strukturierten Mediendaten und
- Multimedia Präsentation Model (MPM), welches eine räumliche und zeitliche Anordnung der unterschiedlich strukturierten Mediendaten und deren Synchronisation erlaubt.

Ausgehend von den rohen Mediendaten, zum Beispiel einem Audiostream, wird im Multimedia Description Model für diese Daten eine Struktur aufgebaut. Dazu werden drei Arten von Objekten definiert, *Kanonische Medienobjekte*, *Medienobjekte* und *Komplexe Objekte*. *Kanonische Medienobjekte* ordnen den rohen Mediendaten eine eindeutige Nummer und medienabhängige Metadaten (z.B. Größe und Farbtiefe eines Bildes) zu. Ein *Medienobjekt* repräsentiert einen Ausschnitt aus einem *Kanonischen Medienobjekt*, der für den Benutzer von besonderem Interesse ist, zum Beispiel Regionen in einem Bild, Frames in einem Videoclip oder Abschnitte in einem Textdokument. Mehrere *Medienobjekte* können zu einem *Komplexen Objekt* zusammengesetzt werden. Das Multimedia Description Model bietet somit eine strukturierte Sicht auf die rohen Mediendaten, wobei verschiedene *Medienobjekte* auf ein und dieselben Mediendaten zugreifen können.

Das Multimedia Interpretation Model bildet den Inhalt der Mediendaten ab. Dazu wird zwischen zwei Ebenen der Darstellung unterschieden, die Feature-Ebene und die Konzept-Ebene. Die Feature-Ebene modelliert für alle Objekte des MDM automatisch erkennbare und meßbare Werte, wie zum Beispiel Farbverteilung, Textur und Gestalt von Bildobjekten. Durch das MIM wird keine konkrete Menge von Features vorgegeben, sondern vielmehr ein allgemeines Framework für die Beschreibung von Features definiert. Ein Feature ist ein Quintupel (*fid*, *dclass*, *extrf*, *simf*, *ftype*), wobei gilt:

- *fid* entspricht der Bezeichnung eines Features, wie zum Beispiel Haarlänge, Augenfarbe, Farbverteilung usw.,
- *dclass* ist die Bezeichnung eines zugeordneten Objektes des Multimedia Description Modells (MDM), für das dieses Feature definiert ist,
- *extrf* gibt einen Algorithmus zur Berechnung des Features aus einem Objekt des MDM an,
- *simf* ist eine Funktion zur Berechnung der Ähnlichkeit zwischen zwei Featurewerten und
- *ftype* definiert den Typ des Features, wie zum Beispiel Histogramm als Typ für die Farbverteilung .

Mit Hilfe dieses Frameworks können entsprechend den Bedürfnissen des Anwenders und abhängig vom Anwendungsgebiet geeignete Features definiert werden. Eine Abbildung räumlicher und zeitlicher Beziehungen läßt sich ebenfalls auf diese Weise realisieren. Auf der Konzept-Ebene werden zu einzelnen Objekten des Multimedia Description Modells Informationen über deren Inhalt abgelegt, welche sich nicht automatisch aus den Mediendaten ableiten lassen, da zusätzliches Hintergrundwissen benötigt wird. Ein Beispiel dafür ist der Name und das Geburtsdatum einer Person auf einem Bildobjekt. Die Modellierung der einzelnen Attribute erfolgt dabei analog zum objektorientierten Ansatz. Beziehungen zwischen den Attributen können zur Abschwächung von Anfrageprädikaten bei einer erfolglosen Suche verwendet werden.

Das Multimedia Präsentation Model erlaubt die räumliche und zeitliche Komposition verschiedener *Medienobjekte* des MDM, um damit eine Präsentation zu erstellen. Das Modell beruht auf zeitabhängigen Petrinetzen [DAPG98], die bereits in Abschnitt 3.3 vorgestellt wurden.

Das Datenmodell von MMSQL bietet alle wesentlichen Merkmale für die Verwaltung von Multimedia-Daten. Durch die verschiedenen Teilmodelle wird schichtweise zuerst die Struktur der Multimedia-Daten und anschließend deren Inhalt modelliert. Das Multimedia Präsentation Modell bildet zusätzlich die Beziehungen für die Präsentation der Multimedia-Daten ab.

Anfragen in MMSQL

Dieser Abschnitt beschreibt die grundlegenden Konzepte der Multimedia Symbolic Query Language (MMSQL). Diese Anfragesprache erweitert die Sprache OQL um die Fähigkeit, unscharfe Anfragen abzubilden. Das Ergebnis einer Anfrage in MMSQL ist eine geordnete Liste von Objekten. Die Ordnung erfolgt anhand der Ranking-Werte, die jedem Ergebnisobjekt zugeordnet sind. Eine Anfrage in MMSQL hat die typische `select-from-where` Struktur:

```
<Q> = select <select-list>
      from <from-list>
      where <condition>
```

`<from-list>` bestimmt als Wurzel der Anfrage die Grundmenge von möglichen Ergebnisobjekten. Folgende Elemente kann die `<from-list>` enthalten: eine Konzept-Klasse des MIM, eine beliebige Klasse des MDM, eine weitere Anfrage oder die Vereinigung, den Durchschnitt, die Differenz, das kartesische Produkt von zwei Anfragen. Die Mengenoperationen entsprechen dabei nicht der Boole'schen Logik, vielmehr kann für die Operanden eine Gewichtung angegeben werden. Zum Beispiel stellt $Union(Q_1, w_1, Q_2, w_2)$ die Vereinigung zweier Anfragen Q_1 und Q_2 dar, mit einer relativen Gewichtung w_1 und w_2 . Das Ergebnis dieser Operation ist die Vereinigung der Ergebnismengen der beiden Anfragen, wobei das Maß für die Ähnlichkeit jedes einzelnen Ergebnisobjektes von dessen Ähnlichkeitsmaß in Q_1 (bzw. Q_2) und der relativen Gewichtung abhängt. Die bei der Kombination verschiedener Rankings auftretenden Probleme und deren Behandlung sind Gegenstand aktueller Forschung der Entwickler dieser Sprache, so daß über die konkrete Semantik der Mengenoperationen an dieser Stelle keine Aussage gemacht werden kann.

`<select-list>` spezifiziert, was für jedes Ergebnisobjekt ausgegeben werden soll. Alle von OQL für die Selektion bekannten Ausdrücke sind auch in MMSQL möglich. Zusätzlich werden einige Konstrukte speziell für das zugrunde liegende Multimedia-Datenmodell angeboten. Dazu gehören beispielsweise der Zugriff auf Featurewerte von abgespeicherten Objekten und die Abfrage von Rankingwerten für Ergebnisobjekte.

`<condition>` dient zur Beschreibung von Prädikaten einer Anfrage. In MMSQL besteht eine Bedingung aus präzisen und unscharfen Prädikaten. Präzise Prädikate beruhen auf Vergleichen zwischen Objekten bzw. Attributen und Werten mit gewöhnlichen Vergleichsoperatoren, wie *gleich*, *kleiner* oder *größer*. Unscharfe Prädikate werden für den Vergleich von Featurewerten, im Zusammenhang mit Rankingwerten und für Anfragen an Objekte der Konzept-Ebene des MIM benötigt. MMSQL bietet hierfür folgende Konstrukte an:

- $Featurewert_1$ **sim** $Featurewert_2$, Ähnlichkeit zwischen zwei Featurewerten,
- $wert$ **in** *Anfrage* überprüft, ob ein Wert zu einer Rankingmenge gehört.

Die vorgestellten einfachen Bedingungen können mit Hilfe der Operatoren *and*, *or* und *not* zu komplexen Bedingungen zusammengesetzt werden. Den zu verknüpfenden Prädikaten wird ähnlich wie bei den Mengenoperationen dabei ein Gewicht zugeordnet. Damit hat der Anwender die Möglichkeit auszudrücken, welches der Prädikate für ihn eine höhere Priorität hat. So kann unter anderem die Relevanz verschiedener Featurewerte vorgegeben werden, zum Beispiel: „Finde alle Bilder, welche ähnlich zum vorgegebenen Beispielbild sind, wobei mehr Wert auf eine ähnliche Farbverteilung als auf die Gestalt gelegt wird.“

Zur Spezifikation der Präsentation von Anfrageergebnissen sind in MMSQL keine Konstrukte vorgesehen, obwohl im zugrunde liegenden Datenmodell die Darstellung der Präsentation berücksichtigt wird. Die Sprachkonstrukte von MMSQL sind hauptsächlich auf die Suche in Multimedia-Daten ausgerichtet.

5.3.2 MOQL und VisualMOQL

Ebenfalls eine Erweiterung der vom ODMG vorgeschlagenen objektorientierten Anfragesprache OQL ist die Multimedia-Object-Query-Language (MOQL). Sie wurde von Li et al. an der Universität von Alberta in Kanada entwickelt und ist in verschiedenen Veröffentlichungen dieser Arbeitsgruppe [LÖS97, LÖSO97, Li98] ausführlich beschrieben. Der Sprachentwurf wurde im Rahmen der Entwicklung des Distributed-Image-Database-Management-Systems (DISIMA) [OÖSI99], eines Prototypen für ein MMDBMS, basierend auf dem Objektdatenbanksystem Objectstore, praktisch erprobt. Aufbauend auf der Sprache MOQL wurde eine visuelle Anfragesprache VisualMOQL entwickelt, welche am Ende dieses Abschnittes beschrieben wird.

MOQL ist ein allgemeiner Ansatz für eine Multimedia-Anfragesprache und konzentriert sich nicht auf einzelne Medien oder spezielle Anwendungen. Hierzu wurde OQL um räumliche und zeitliche Anfragekonstrukte erweitert. Außerdem kann mit Hilfe von MOQL die Präsentation der Anfrageergebnisse festgelegt werden. Bevor die einzelnen Konstrukte von MOQL untersucht werden, befaßt sich der nachfolgende Abschnitt zunächst mit dem Multimedia-Datenmodell, welches eine Menge von Multimedia-Grundobjekten und dazugehörigen Funktionen definiert.

Das DISIMA-Datenmodell

Das der Sprache MOQL zugrunde liegende Datenmodell definiert als Teil des DISIMA-Systems Konzepte zur Verwaltung der Multimedia-Daten und den damit verbundenen Metadaten, um eine inhaltsbasierte Suche zu ermöglichen. Das objektorientierte Modell, welches in Abbildung 5.5 dargestellt ist, setzt sich aus zwei Blöcken zusammen, dem Primitiv-Media-Block und dem Salient-Object-Block. Ein Block ist hierbei eine Gruppe semantisch zusammengehörender Entities.

Der Primitiv-Media-Block besteht aus zwei Ebenen, der reinen Medienebene (Primitiv-Media-Layer) für den Zugriff auf die rohen Mediendaten und der Darstellungsebene

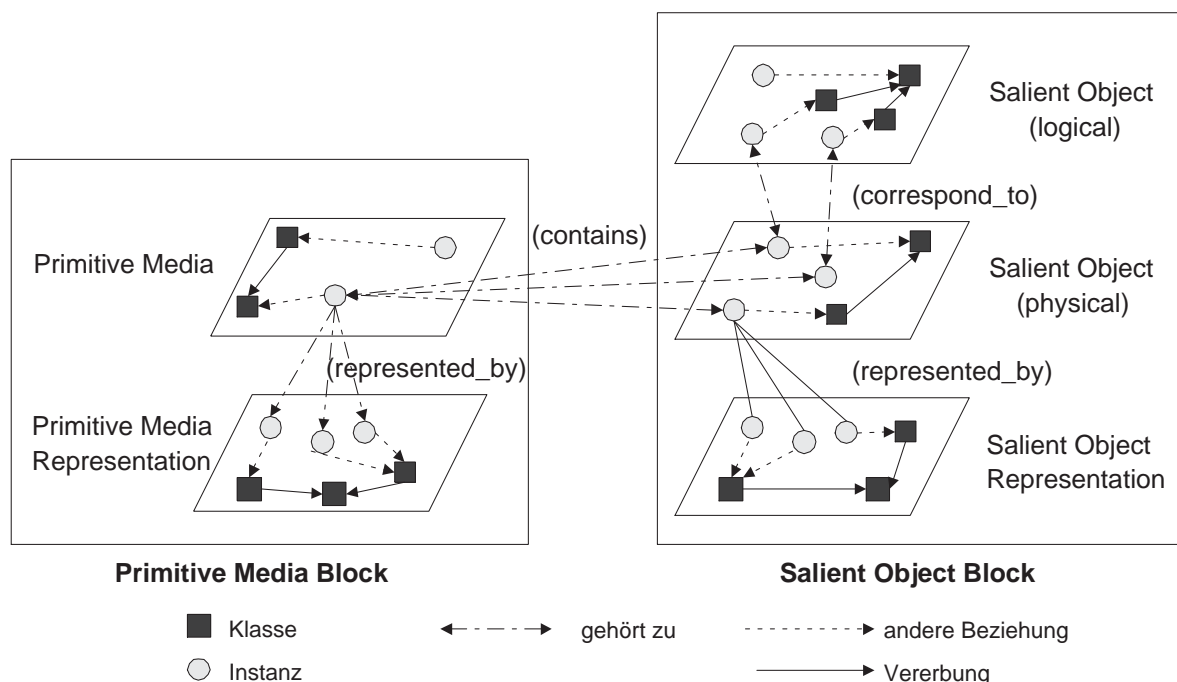


Abbildung 5.5: Überblick über das DISIMA-Datenmodell

der reinen Mediendaten (Primitive-Media-Representation-Layer), um verschiedene Darstellungen der reinen Mediendaten abzubilden. Diese Unterscheidung wird vorgenommen, um die eigentlichen Multimedia-Daten und die Art ihrer Darstellung voneinander zu trennen.

Der Salient-Object-Block modelliert den Inhalt der Medien. Dieser wird dargestellt als eine Menge von wesentlichen Objekten (Salient Object), welche über zeitliche und räumliche Beziehungen miteinander in Beziehung stehen. Dabei wird zwischen physischen und logischen Objekten unterschieden.

Die physischen Objekte werden aus den Mediendaten abgeleitet. Dabei können mehrere Objekte in einem Medium enthalten sein, wie zum Beispiel mehrere Personen auf einem Bild. Räumliche Beziehungen zwischen den einzelnen physischen Objekten können durch topologische und Richtungsbeziehungen angegeben werden, wie sie im wesentlichen in Abschnitt 3.2.2 vorgestellt wurden. Die Darstellung zeitlicher Abhängigkeit wird durch die Unterstützung von Allens 13 Intervallbeziehungen (vgl. Abschnitt 3.3.1) erreicht. Für die einzelnen physischen Objekte können ähnlich wie für die reinen Mediendaten mehrere Darstellungen existieren. Zum Beispiel ist für eine Person auf einem Bild eine Darstellung sowohl durch ein minimales umschreibendes Rechteck als auch durch eine Punktmenge möglich. Im DISIMA-Datenmodell erfolgt die Abbildung der verschiedenen Darstellungen auf der Salient-Object-Representation-Ebene.

Ein logisches Objekt (logical salient Object) interpretiert mit Unterstützung des Benutzers die in ein oder mehreren physischen Objekten enthaltenen Informationen. Dabei

kann ein physisches Objekt, abhängig von der Anwendung, auch mehrere Interpretationen besitzen. Die Objekte der logischen Ebene können über eine Spezialisierungshierarchie miteinander verknüpft sein.

Durch die Unterteilung in Ebenen ist eine unterschiedliche Repräsentation der Medienobjekte möglich, und es wird eine Trennung von Semantik und physischen Merkmalen erreicht. So benötigt die Anfrage: „Finde alle Videos, welche den Bundeskanzler zeigen.“, ausschließlich Informationen der logischen Objektebene, wogegen die Anfrage: „Finde alle Videos, wo der Bundeskanzler rechts vom Außenminister steht.“, Informationen sowohl der logischen Objektebene als auch der physischen Objektebene benötigt.

Anfragen in MOQL

Die Multimedia-Anfragesprache MOQL bietet sehr umfangreiche Möglichkeiten, um Anfragen zu formulieren. Es können räumliche, zeitbezogene und inhaltsbasierte Anfragen medienübergreifend gestellt werden. Die Spezifikation der Präsentation von Anfrageergebnissen bei der Anfrageformulierung ist ebenfalls möglich. Dahingegen finden unscharfe Anfragemethoden kaum eine Unterstützung. Ein Ähnlichkeitsoperator für räumliche Anfragen wird zwar angeboten, das Ergebnis einer solchen Operation ist aber lediglich ein Boole'scher Wert, wodurch ein Ranking von ähnlichen Objekten nicht möglich ist. Eine Anfrage in MOQL hat die folgende allgemeine Syntax:

```

select      [distinct] <projection_attributes>
from        <query> [ [as] <identifizier>]
            {, <query> [ [as] <identifizier>] }
[where     <query>]
[present   <layout> { and <layout>}]
```

Neben dem Hinzufügen einer **present**-Klausel erweitert MOQL vor allem die **where**-Klausel der Anfragesprache OQL. Hierfür werden drei neue Ausdrücke definiert:

- `<spatial_expression>` zur Beschreibung räumlicher Beziehungen,
- `<temporal_expression>` zur Beschreibung zeitlicher Beziehungen und
- `<contain_predicate>` zur Beschreibung von Beziehungen zwischen abgeleiteten Objekten und den reinen Mediendaten.

`<spatial_expression>` können unterteilt werden in räumliche Prädikate und räumliche Funktionen, die für die Formulierung einer Anfrage angeboten werden. Räumliche Prädikate vergleichen die räumlichen Eigenschaften der Datenbankobjekte und liefern einen Boole'schen Wert als Ergebnis. Eine sehr umfangreiche Menge von Prädikaten steht hierfür zur Verfügung. Darin enthalten sind neben den in Abschnitt 3.2.2 vorgestellten topologischen Beziehungen und den Richtungsbeziehungen auch Prädikate zum

Vergleich ganzer Bilder, wie **identical** (Test, ob zwei Bilder bezüglich enthaltener Objekte, Farbverteilung und Textur identisch sind), **subpicture** (Überprüfung, ob ein Bild in einem anderen enthalten ist) oder **similar** (Ähnlichkeitsvergleich zwischen zwei Bildern, der einen Boole'schen Wert berechnet).

Die angebotenen räumlichen Funktionen berechnen Attribute eines Objektes oder als Aggregatfunktionen Werte auf einer Menge von räumlichen Objekten. MOQL bietet hierfür ebenfalls eine umfangreiche Auswahl von Funktionen an, wie zum Beispiel **distance** (räumlicher Abstand zweier Objekte) oder **area** (räumliche Fläche eines Objektes). Die Verwendung räumlicher Prädikate und Funktionen in einer Anfrage soll an einem Beispiel verdeutlicht werden. Die Anfrage: „Suche alle Waldgebiete, die in Sachsen-Anhalt liegen und berechne die Größe dieser Gebiete.“, ist nachfolgend in MOQL formuliert.

```
select wald, area(wald.region)
from Waldgebiet wald, Bundesland land
where wald.region inside land.region
and land.name.equal(„Sachsen-Anhalt“)
```

Die obige Anfrage enthält das räumliche Prädikat **inside**, durch welches alle Regionen ausgewählt werden, die ein Waldgebiet enthalten. Anschließend wird durch die räumliche Funktion **area** die Fläche der Waldgebiete berechnet.

<temporal_expression> setzen sich ebenfalls zusammen aus zeitlichen Prädikaten und Funktionen. Als zeitbezogene Datentypen unterstützt MOQL Zeitpunkte, Zeitspannen und Zeitintervalle, welche ähnlich wie in Abschnitt 3.3.1 definiert sind. Als Vergleichsprädikate für Zeitintervalle werden die 13 allgemein benutzten Zeitintervallbeziehungen von Allen [All83] angeboten.

Das Angebot an zeitbezogenen Funktionen in MOQL umfaßt unäre Intervallfunktionen zur Berechnung der Intervalllänge und der Intervallgrenzen, Mengenoperationen auf Zeitintervallen, Vergleichsoperationen für Zeitpunkte und Zeitspannen und viele weitere. Hervorgehoben werden müssen die Continuous-Media-Funktionen, von denen MOQL im wesentlichen die Auswahl von Ausschnitten und Frames in Videofilmen (**first_Frame**, **last_Frame** usw.) und Boole'sche Funktionen für den Test auf bestimmte Kamerabewegungen unterstützt, wie zum Beispiel **zoomIn**, **panLeft**, **cut** oder **fade**.

<contain_predicate> drücken Bedingungen aus, mit deren Hilfe überprüft werden kann, ob ein abgeleitetes Objekt (salient object) in einem bestimmten Medium enthalten ist oder nicht. Zum Beispiel drückt folgende Anfrage die Suche nach Bildern aus, auf denen Personen zu sehen sind.

```
select m
from Bilder m, Personen p
wherem contains p
```

Für die Präsentation der Anfrageergebnisse hat MOQL zusätzlich eine **present**-Klausel eingeführt. In dieser lassen sich das räumliche Layout (z.B. Position und Größe des

Ausgabefensters), der zeitliche Ablauf (z.B. Startzeitpunkt, Dauer und Synchronisation der Präsentation einzelner Medienobjekte) und ein Szenario Layout, eine Art Framework zur Einbindung anderer Präsentationsmodelle und -sprachen, spezifizieren. Zur Unterstützung der Anforderungen einer Ergebnispräsentation sind in MOQL folgende Funktionen definiert:

- **atWindow** (*id*, *Punkt_1*, *Punkt_2*), ein Medienobjekt mit der Bezeichnung *id* wird in einem Fenster, das durch *Punkt_1* und *Punkt_2* aufgespannt wird, dargestellt.
- **atTime** (*Startzeitpunkt*) definiert den *Startzeitpunkt* für die Präsentation.
- **display** (*id*, *Startzeitpunkt*, *Dauer*), ein non-continuous Medienobjekt wird zum *Startzeitpunkt* für eine bestimmte *Dauer* angezeigt.
- **play** (*id*, *Startzeitpunkt*, *Dauer*, *Geschwindigkeit*), ein continuous Medienobjekt wird zum *Startzeitpunkt* für eine bestimmte *Dauer* mit einer bestimmten *Geschwindigkeit* abgespielt.
- **thumbnail** (*id*), ein Medienobjekt wird als Thumbnail präsentiert.
- **resize** (*id*, *Höhe*, *Breite*) legt die *Höhe* und *Breite* eines Medienobjektes für die Präsentation fest.
- **parStart** startet zwei Präsentationskomponenten gleichzeitig.
- **parEnd** beendet eine Präsentationskomponente gleichzeitig mit einer anderen.
- **after** startet eine Präsentationskomponente, nach dem die andere beendet ist.

Zum Abschluß der Ausführungen über die Multimedia-Anfragesprache MOQL soll ein Beispiel die Verwendung der Präsentationskonstrukte verdeutlichen:

```

select  m, v
from    Bilder m, Videos v, Personen p
where   m contains p or
        v contains p and
        p.name.equal(„Karl Marx“)
present atWindow (m, (0,0), (320,240)) and
        atWindow (v, (321,0), (641,240)) and
        play (v, 10, 30*60, normal) parStart
        display (m, 0, 20)

```

Durch die obige Anfrage werden alle Bilder und Videos ausgewählt, die Karl Marx zeigen. Die Darstellung der Bilder erfolgt in einem Fenster mit den Koordinaten (0,0), (320,240) und die Videos werden in einem Fenster mit den Koordinaten (321,0),

(641, 240) gezeigt. Die Bilder werden zehn Sekunden vor dem Start des Videos für eine Dauer von 20 Sekunden angezeigt. Die Abspieldauer eines Videos beträgt 30 Minuten und die Abspielgeschwindigkeit wird mit „normalt“ angegeben. Sowohl in der „where“-Klausel als auch in der „present“-Klausel wird dasselbe Schlüsselwort **and** verwendet, wobei sich aber die Bedeutung unterscheidet. Während in der „where“-Klausel hiermit die logische „UND“-Verknüpfung zweier Terme gemeint ist, dient es in der „present“-Klausel lediglich als Bindeglied zwischen zwei Präsentationsangaben.

VisualMOQL

Eine visuelle Anfragesprache, basierend auf der oben vorgestellten Multimedia-Anfragesprache MOQL, ist VisualMOQL [OXÖ99]. Dabei unterstützt VisualMOQL in der derzeitigen Version lediglich Bildanfragen. Eine Anfrage wird für die Verarbeitung auf die Sprache MOQL abgebildet. Die räumlichen Prädikate und Funktionen von VisualMOQL entsprechen somit den bereits im Zusammenhang mit MOQL vorgestellten und auch das zugrunde liegende Datenmodell ist dasselbe.

VisualMOQL wurde entwickelt, um die Vorteile einer visuellen Anfragetechnik mit der Ausdrucksfähigkeit einer textbasierten Anfragesprache zu verbinden. Nachfolgend sind einige wesentliche Merkmale der Sprache aufgeführt:

- Aufbau einer Anfrage Schritt für Schritt, das heißt, zuerst wird eine Bildkategorie ausgewählt, anschließend werden einfache Suchbedingungen definiert und diese dann zu komplexen Bedingungen zusammengesetzt,
- Visuelle Beschreibung von Anfrageprädikaten, wie Farbe, Gestalt, Textur und räumliche Beziehungen, und
- Verfeinerung der Anfrageprädikate in jeder Phase des Anfrageprozesses.

Da VisualMOQL nur Bildanfragen unterstützt, beschränken sich die Präsentationsfähigkeiten auch auf Bilddaten. Bei der Anfragespezifikation wird die maximale Anzahl der Bilder angegeben, die zurückgeliefert werden sollen. Durch die Ergebnismenge kann anschließend navigiert werden und zugeordnete Attribute lassen sich anzeigen.

5.4 Weitere Ansätze

Im Verlauf dieses Kapitels wurde bereits eine Reihe von Ansätzen für Multimedia-Anfragesprachen vorgestellt. Dennoch läßt sich nicht der Anspruch auf Vollständigkeit erheben, da durch zahlreiche Forschungsarbeiten auf dem Gebiet der Multimedia-Datenbanken ständig neue Anfragesprachen entwickelt werden. Des weiteren lassen sich, eingeschränkt durch den zeitlichen Rahmen dieser Arbeit, nicht alle Sprachansätze ausführlich betrachten. Es wurde vielmehr versucht, solche Sprachansätze für eine umfangreiche Betrachtung auszuwählen, die das meiste Potential für eine Verwendung als

Anfragesprache in einer Multimedia-Datenbank aufweisen. Im folgenden werden einige der bisher noch nicht betrachteten Ansätze für Multimedia-Anfragesprachen kurz vorgestellt. Außerdem wird auf Sprachansätze in artverwandten Gebieten kurz eingegangen.

Eine der Anfragesprachen die bisher noch nicht betrachtet wurden ist MQL [LC98, HK96] (Multimedia Query Language). MQL basiert auf einem objektorientierten Framework zur Verwaltung von Multimedia-Daten und unterstützt komplexe Objktanfragen, verfeinernde Anfragen, geschachtelte Anfragen und Prädikate für Ähnlichkeitsanfragen an Bilder, Texte und Sprachdokumente. Die Anfragesprache basiert auf SQL, wobei das Sprachdesign sehr ähnlich zu der in Abschnitt 5.2.2 untersuchten Sprache CSQL ist. Beide Sprachen haben unter anderem das *contain*-Prädikat gemeinsam. Im Vergleich zu MQL bietet CSQL aber einige zusätzliche Prädikate (*is_a*, *s_like*) an.

Hirzalla und Karmauch stellen in [HK96] eine Multimedia Query Spezifikation Language (MMQSL) vor. Das Modell zu dieser Sprache beruht auf der Zerlegung von Multimedia-Dokumenten in mehrere Multimedia-Segmente. Die einzelnen Multimedia-Segmente enthalten räumliche und zeitliche Informationen sowie medienspezifische Merkmale, wie z. B. Kamerabewegungen in Videodaten oder Farbverteilung in Bildern. Anfragen werden an Mengen von Multimedia-Dokumenten gestellt. Eine Anfrage in MMSQL hat die folgende allgemeine Form:

```

find X in Doc-list
where
  begin
    [Temporale-Spezifikation] and | or
    [Räumliche-Spezifikation] and | or
    [Medienspezifische-Spezifikation] and | or
    [Externe-Spezifikation]
  end

```

Externe-Spezifikationen sind dabei Angaben zu möglichen Benutzerinteraktionen während der Präsentation. Für die Formulierung von Anfragen werden unzählige Prädikate für sämtliche Multimedia-Datentypen angeboten, wobei sich die Autoren auf die Beschreibung der Syntax beschränken. Unscharfe Anfragen oder das Schachteln von Anfragen ist in MMSQL nicht möglich.

Eine sehr wichtige Anforderung an Anfragesprachen für Multimedia-Datenbanken ist, wie in Abschnitt 3.5 erläutert, die Unterstützung der Präsentation von Anfrageergebnissen. Ausschließlich dieser Aufgabe widmet sich die Anfragesprache SQL+D [BGN98, BGS98]. Sie erweitert SQL3 um Möglichkeiten zur Präsentationsunterstützung für Multimedia-Daten. Hierfür wird eine zusätzliche *display*-Klausel eingeführt. In dieser Klausel lassen sich die räumliche und zeitliche Darstellung der Anfrageergebnisse und mögliche Benutzerinteraktionen spezifizieren.

Für die inhaltsbasierte Suche in Multimedia-Daten existieren einige weitere Systeme, die zwar Anfragen erlauben, hierfür aber keine eigene Anfragesprache besitzen. Die entsprechende Funktionalität wird bei solchen Systemen über grafische Benutzeroberflächen

angeboten. Während sich ad-hoc Anfragen auf diese Art und Weise sehr komfortabel formulieren lassen, ist eine Einbindung in Programmiersprachen nicht möglich. Nachfolgend sind einige Beispiele für solche Systeme aufgeführt:

- *QBIC* [FSN⁺97], ein System zur inhaltsbasierten Suche in Bild und Videodaten anhand der Gestalt, Farbverteilung, Textur und weiteren aus den Daten automatisch abgeleiteten Informationen. Der Inhalt der gesuchten Bilder wird vom Anwender durch Skizzen vorgegeben.
- *VisualSEEK* [SC97], ähnlich wie QBIC bietet dieses System eine inhaltsbasierte Bildsuche mittels abgeleiteter Bildmerkmale an.
- *VideoSTAR* [Hje95], ein experimentelles Datenbankframework für die Verwaltung von Videodaten und der Möglichkeit zur Formulierung von komplexen zeitlichen Anfragen.

Abschließend sollen in diesem Kapitel Anfragesprachen erwähnt werden, die zwar auf anderen Gebieten der Datenbankforschung entwickelt wurden, welche aber dennoch einige typische Anforderungen an Multimedia-Anfragesprachen erfüllen. So wurden im Zusammenhang mit temporalen Datenbanken zahlreiche Anfragesprachen entwickelt. Diese ermöglichen Anfragen an Zeitpunkte und Zeitintervalle und bieten grundlegende zeitbezogene Operationen an, wie sie auch in MMDBMS benötigt werden. Ein Überblick über temporale Datenbanken und entsprechende Anfragesprachen gibt [ZCF⁺97]. TSQL2, ein prominenter Vertreter dieser Sprachen, wird in [Myr97] vorgestellt. Ein generelles Problem von temporalen Anfragesprachen im Zusammenhang mit Multimedia-Daten ist, daß sich temporale Anfragen hauptsächlich auf die Historie der Daten (Zustände der Daten zu verschiedenen Zeitpunkten) beziehen. In einem MMDBMS müssen aber zusätzlich Continuous-Datentypen durch Anfragen unterstützt werden, was mit temporalen Anfragesprachen im allgemeinen nicht möglich ist.

Im Zusammenhang mit Geographischen Informationssystemen und CAD-Anwendungen wurden verschiedene Forschungsarbeiten über die Integration räumlicher Daten in ein DBMS veröffentlicht. Dabei wurde unter anderem die Anfragesprache Pictorial Query-By-Example [PS95, DSS98] (PQBE) vorgeschlagen. Für Anfragen in dieser Sprache werden in ein Gitter, welches die verschiedenen Bildregionen repräsentiert, Richtungsbeziehungen und topologische Beziehungen eingetragen. Anhand der hierdurch vorgegebenen Bildstruktur wird eine Ähnlichkeitssuche durchgeführt. Dieser Ansatz bildet die Grundlage für einige grafische Anfragesprachen (z. B. QBIC) in MMDBMS.

6 | Vergleich und Bewertung

In Kapitel 5 wurde eine Reihe von Ansätzen für Multimedia-Anfragesprachen vorgestellt. Die grundlegenden Konzepte der einzelnen Anfragesprachen unterscheiden sich dabei sehr stark voneinander. Um eine Einschätzung der Leistungsfähigkeit der einzelnen Sprachen vornehmen zu können, werden diese anhand ihrer Eigenschaften und den in Kapitel 4 aufgeführten Kriterien für Multimedia-Anfragesprachen im nachfolgenden Abschnitt gegenübergestellt und miteinander verglichen. Aufbauend auf diesem Vergleich werden im zweiten Abschnitt dieses Kapitels die verschiedenen Sprachansätze klassifiziert. Daran anschließend erfolgt eine Untersuchung, inwieweit die vorgestellten Sprachansätze für den Einsatz in Multimedia-Datenbank-Management-Systemen geeignet sind. Am Ende dieses Kapitels werden Entwurfsprinzipien für zukünftige Entwicklungen von Multimedia-Anfragesprachen vorgeschlagen.

6.1 Vergleich von Multimedia-Anfragesprachen

In diesem Abschnitt werden die Vor- und Nachteile der einzelnen Multimedia-Anfragesprachen aufgezeigt. Dazu erfolgt eine Gegenüberstellung und ein Vergleich der Sprachansätze, basierend auf den in Kapitel 4 aufgestellten Kriterien. Dabei werden zunächst die allgemeinen Kriterien betrachtet, welche für jede Anfragesprache gelten. Daran anschließend erfolgt eine Betrachtung der multimediaspezifischen Kriterien in bezug auf die Anfragesprachen.

Allgemeine Kriterien

Allgemeine Kriterien für Multimedia-Anfragesprachen wurden bereits in Abschnitt 4.1 erläutert. Diese sollen nun für die Untersuchung der Sprachansätze benutzt werden. Dabei wird die Erfüllung der einzelnen Kriterien durch die jeweilige Anfragesprache überprüft. Der Abschnitt gliedert sich hierfür in zwei Teile. Im ersten Teil werden die SQL/OQL-basierten Sprachansätze und im zweiten Teil die neu entwickelten Ansätze betrachtet.

Ein Großteil der vorgestellten Multimedia-Anfragesprachen basiert auf einer existierenden objektorientierten oder relationalen Anfragesprache. Tabelle 6.1 enthält eine Gegenüberstellung dieser Sprachansätze. In der ersten Spalte sind die verschiedenen

allgemeinen Kriterien aufgeführt. Die anderen Spalten geben jeweils die Gültigkeit des Kriteriums für die einzelnen Multimedia-Anfragesprachen an. Ein „√“ bedeutet dabei, daß das Kriterium durch die Anfragesprache erfüllt wird und ein „(√)“ kennzeichnet ein nur teilweise erfülltes Kriterium. Das Fragezeichen gibt an, daß über die Erfüllung des Kriteriums keine Aussage gemacht werden kann. Ist ein Kriterium nicht erfüllt enthält die Tabelle ein „–“.

Kriterium	KEQL	CSQL	MADT	VideoSQL	MMSQL	MOQL
Ad-hoc Formulierung	√	√	√	√	√	√
Generische Operatoren	√	√	√	√	√	√
Anwendungsunabhängigkeit	(√)	√	√	√	√	√
Deskriptivität	√	√	√	√	√	√
Mengenorientiertheit	√	√	√	?	√	√
Orthogonalität	–	√	√	–	√	√
Effizienz	?	(√)	(√)	?	(√)	(√)
Erweiterbarkeit	√	√	√	–	√	√
Abgeschlossenheit	√	√	√	√	√	√
Adäquatheit	√	√	√	√	–	√
Vollständigkeit	(√)	√	√	–	√	√
Sicherheit	√	(√)	(√)	√	(√)	(√)
Optimierbarkeit	√	(√)	(√)	?	(√)	(√)
Formale Semantik	√	√	√	–	√	√
Eingeschränktheit	√	–	–	√	–	–

Tabelle 6.1: Vergleich SQL/OQL-basierter Sprachansätze anhand allgemeiner Kriterien

Durch die zugrunde liegende Standardsprache sind bereits wesentliche Eigenschaften vorgegeben, die sich auch bei einer Erweiterung der Anfragesprache nicht verändern. So erfüllen die SQL-basierten Anfragesprachen (KEQL, CSQL und MADT) fast alle allgemeinen Kriterien. Einige Kriterien werden aber auch nur unzureichend oder gar nicht erfüllt. Diese sind im einzelnen:

- *Orthogonalität*: In SQL und demzufolge auch in darauf aufbauenden Anfragesprachen wird dieses Kriterium dadurch verletzt, daß Anfrageterme nicht an allen Stellen (z.B. nach der from Klausel) möglich sind. Dies gilt allerdings nicht für Sprachansätze, welche auf dem SQL3-Standard beruhen, was für die meisten in dieser Arbeit untersuchten Ansätze (CSQL, MADT) der Fall ist.
- *Eingeschränktheit*: Anfragen in SQL3 können Methodenaufrufe enthalten. Da diese Methoden in einer Programmiersprache entworfen werden, ist das Kriterium der Eingeschränktheit durch die Berechnungsvollständigkeit der Programmiersprache nicht erfüllt. Die negativen Auswirkungen auf die Effizienz, Sicherheit und Opti-

mierbarkeit werden zwar aufgrund der Kontrolle der Methodenausführung durch das DBMS abgeschwächt, bleiben aber grundsätzlich bestehen.

- *Anwendungsunabhängigkeit*: Die Anfragesprache KEQL wurde im Hinblick auf medizinische Anwendungen entwickelt und ist deshalb nur teilweise anwendungsunabhängig.

Gesondert betrachtet werden muß die Anfragesprache VideoSQL. Für diese Sprache gelten die obigen Ausführungen nicht, da VideoSQL lediglich den bekannten „SFW“-Block übernimmt, sonst aber kaum etwas mit SQL gemeinsam hat. VideoSQL unterstützt eine ad-hoc Formulierung und generische Operatoren, ist anwendungsunabhängig, deskriptiv, abgeschlossen und eingeschränkt. Da auch vom Datenmodell nur Videodaten und Videosequenzen als Anfrageergebnisse unterstützt werden, ist die Adäquatheit und Abgeschlossenheit trivialerweise erfüllt. Anhand der über VideoSQL zur Verfügung stehenden Literatur kann keine Aussage über die Mengenorientiertheit, Effizienz und Optimierbarkeit getroffen werden, was in Tabelle 6.1 durch ein Fragezeichen ausgedrückt wird. VideoSQL erfüllt die Kriterien Orthogonalität und Erweiterbarkeit nicht. Des Weiteren existiert keine formale Semantik, und die Vollständigkeit bezüglich der relationalen Algebra ist nicht gegeben.

Für OQL-basierte Multimedia-Anfragesprachen, wie MMSQL und MOQL, gelten im wesentlichen die gleichen Aussagen, die für SQL-basierte Anfragesprachen getroffen wurden. Die Eingeschränktheit wird auch hier nicht erfüllt. Damit verbunden sind Auswirkungen auf die Effizienz, Sicherheit und Optimierbarkeit, da das Ergebnis, die Laufzeit und die Terminierung einer Methodenabarbeitung nicht vorhersehbar sind. Gegenüber SQL erfüllt OQL aber das Kriterium der Orthogonalität, denn in OQL-Anfragen können Operationen entsprechend dem Typsystem und den Ein- und Ausgangstypen frei miteinander kombiniert werden. Dies gilt natürlich ebenso für die beiden in dieser Arbeit vorgestellten OQL-basierten Multimedia-Anfragesprachen MMSQL und MOQL. Während von MOQL außer den obigen Einschränkungen alle weiteren allgemeinen Kriterien für eine Anfragesprache erfüllt werden, verletzt MMSQL zusätzlich die Adäquatheit. Denn obwohl das Datenmodell eine Präsentation der Anfrageergebnisse unterstützt, bietet die Anfragesprache hierfür keine Konstrukte an.

Neben Multimedia-Anfragesprachen, die auf Standardsprachen aufbauen, wurden in Kapitel 5 weitere Sprachansätze vorgestellt, welche im Zusammenhang mit neu entwickelten Multimedia-Datenbank-Management-Systemen entstanden sind. Diese sollen nun anhand der allgemeinen Kriterien verglichen werden. Hierfür enthält Tabelle 6.2 eine Gegenüberstellung der neu entwickelten Sprachansätze. Der Aufbau dieser Tabelle entspricht dem von Tabelle 6.1, welche die allgemeinen Kriterien den SQL/OQL-basierten Sprachansätzen gegenüberstellt. Ein „√“ kennzeichnet wiederum ein erfülltes und ein „-“ ein unerfülltes Kriterium.

Ein Problem beim Vergleich der neu entwickelten Sprachansätze ist die größtenteils unvollständige Sprachbeschreibung in der Literatur, welche für die verschiedenen Ansätze zur Verfügung steht. Deshalb konnten einige Kriterien nicht für alle neu ent-

Kriterium	PICQUERY+	Score	TVQL	SMDS
Ad-hoc Formulierung	✓	✓	✓	✓
Generische Operatoren	✓	✓	✓	✓
Anwendungsunabhängigkeit	✓	–	✓	✓
Deskriptivität	✓	✓	✓	✓
Mengenorientiertheit	✓	✓	✓	✓
Orthogonalität	–	–	–	–
Effizienz	?	?	?	✓
Erweiterbarkeit	–	–	–	✓
Abgeschlossenheit	✓	✓	✓	✓
Adäquatheit	✓	✓	✓	✓
Vollständigkeit	?	?	?	✓
Sicherheit	?	?	?	✓
Optimierbarkeit	?	?	?	✓
Formale Semantik	?	?	?	✓
Eingeschränktheit	✓	✓	✓	✓

Tabelle 6.2: Vergleich neu entwickelter Sprachansätze anhand allgemeiner Kriterien

wickelten Sprachansätze auf ihre Gültigkeit untersucht werden. In den Fällen, wo eine Einschätzung nicht möglich war, enthält die Tabelle 6.2 an der entsprechenden Stelle ein Fragezeichen.

Fast alle der neu entwickelten Systeme (bis auf die logikbasierte Anfragesprache des SMDS) bieten für die Formulierung von Anfragen ausschließlich eine grafische Anfragesprache bzw. eine grafische Benutzeroberfläche an. Diese erfüllen durch fest vorgegebene grafische Anfragekonstrukte die Forderung nach einer ad-hoc Formulierung, generischen Operatoren und Deskriptivität. Lediglich in SCORE ist die Anwendungsunabhängigkeit unzureichend, weil für die Formulierung von Anfragen eine Grundmenge markanter Icons (vgl. Abschnitt 5.1.2) benutzt wird, welche abhängig vom Anwendungsgebiet fest vorgegeben sind. Da die Auswahl der Ergebnisobjekte in allen vorgestellten Sprachansätzen auf Mengen von Objekten basiert, wird das Kriterium der Mengenorientiertheit von diesen Anfragesprachen erfüllt.

Die Orthogonalität fordert eine beliebige Kombination von Operationen. Dies kann von den untersuchten Sprachansätzen aufgrund der durch die grafischen Methoden fest vorgegebenen Struktur einer Anfrage nicht erfüllt werden. So ist beispielsweise eine Schachtelung oder Kopplung von Tabellengerüsten in PICQUERY+ nicht möglich.

Über eine effiziente Ausführung der Basisoperationen kann keine eindeutige Aussage gemacht werden, da hierfür der Aufwand zur Ausführung der Operationen der jeweiligen Sprache bekannt sein müßte. Da aber die vorgestellten Multimedia-Anfragesprachen keine benutzerdefinierten Methoden unterstützen, sollte im allgemeinen eine effiziente Realisierung der Anfragen möglich sein. Für die logikbasierte Anfragesprache des SMDS

ist das Kriterium der Effizienz erfüllt, da in [MS96] Indexstrukturen und geeignete Algorithmen mit akzeptabler linearer Komplexität für die Ausführung von Anfragen auf dieser Struktur beschrieben werden.

Ein Kriterium, das ebenfalls nur durch das SMDS erfüllt wird, ist die Erweiterbarkeit. Das hierfür vorgeschlagene Framework erlaubt die Erweiterung um zusätzliche Medientypen und Features sowie eine Benutzung dieser Erweiterungen in Anfragen. Die restlichen Sprachansätze unterstützen keine solche Erweiterung.

Die Anfrageergebnisse von PICQUERY+, SCORE und TVQL bestehen aus Bild- und Videodaten und alphanumerischen Daten, welche in dem jeweiligen zugrunde liegenden Datenmodell darstellbar sind. Somit ist die Abgeschlossenheit dieser Sprachen gewährleistet. Auch die Adäquatheit wird von den obigen Anfragesprachen erfüllt, denn alle Konstrukte der teilweise recht einfachen Datenmodelle können durch die dazugehörige Anfragesprache ausgenutzt werden. Die logikbasierte Anfragesprache des SMDS erfüllt ebenfalls die Abgeschlossenheit und die Adäquatheit, denn einerseits läßt die Sprachdefinition nur Medienobjekte, Features und Attributwerte der SMDS sowie die leere Menge als Anfrageergebnis zu (Abgeschlossenheit), andererseits werden alle Konzepte der strukturierten Multimedia-Datenbank durch die logikbasierte Anfragesprache unterstützt (Adäquatheit).

Außer für das Framework einer strukturierten Multimedia-Datenbank (SMDS), bei der die logikbasierte Anfragesprache selbst eine Art Anfragekalkül darstellt, wurde für keine der anderen Multimedia-Anfragesprachen in den entsprechenden Veröffentlichungen eine formale Semantik angegeben. Ob eine solche nicht existiert oder in den Veröffentlichungen nur nicht beschrieben wurde, konnte im Rahmen dieser Arbeit nicht festgestellt werden. Da eine formale Semantik Voraussetzung für eine Optimierung und die Entscheidung über die Vollständigkeit einer Anfragesprache ist, kann an dieser Stelle keine Aussage über die Erfüllung dieser Kriterien getroffen werden.

Ebenfalls aufgrund einer unvollständigen Semantikfestlegung schwer zu entscheiden, ist das Kriterium der Sicherheit. Dieses fordert für syntaktisch korrekte Anfragen endliche Ergebnisse und eine Terminierung dieser Anfragen in endlicher Zeit. Lediglich für die logikbasierte Anfragesprache an SMDS wird in [MS96] hierfür ein formaler Beweis angegeben.

Da alle hier betrachteten Multimedia-Anfragesprachen nicht berechnungsvollständig sind, ist das Kriterium der Eingeschränktheit für alle diese Sprachen erfüllt.

Multimediaspezifische Kriterien

Nachdem im vorhergehenden Abschnitt die einzelnen Sprachansätze in Hinblick auf die Erfüllung der allgemeinen Kriterien untersucht wurden, soll nun festgestellt werden, inwieweit die multimediaspezifischen Anforderungen durch die im vorhergehenden Kapitel vorgestellten Ansätze für eine Multimedia-Anfragesprache erfüllt werden. Grundlage für diese Untersuchung sind die in Abschnitt 4.2 betrachteten multimediaspezifischen Kriterien.

Anfragesprache	Universell				Inhaltsanfragen			Raum				Zeit		Unscharfe Anfragen				Präsentation				
	V	B	T	A	F	S	AV	R	T	A	M	P	O	P	G	Ä	R	R	Z	TF	TL	Q
Picquery+	-	√	√	-	√	√	-	√	√	√	-	√	√	√	-	√	√	√	√	√	-	-
SCORE	-	√	-	-	-	√	√	√	√	√	-	-	-	-	-	-	√	√	-	-	-	-
TVQL	√	-	-	-	-	√	√	√	√	-	-	√	-	-	-	-	-	√	√	-	-	-
SMDS	√	√	√	√	√	√	√	√	√	√	-	√	√	√	-	√	-	-	-	-	-	-
KEQL	-	√	√	-	√	√	√	√	√	√	-	√	-	√	-	√	√	√	√	√	-	-
CSQL	-	√	√	-	√	√	√	√	-	-	-	-	-	-	-	√	-	√	-	-	-	-
MADT	√	√	√	√	√	√	-	√	√	√	√	√	√	-	√	√	√	-	-	√	-	-
VideoSQL	√	-	-	-	-	√	-	√	√	-	-	√	-	-	-	-	-	-	√	-	-	-
MMSQL	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	-	-	-	-	-
MOQL	√	√	√	√	√	√	-	√	√	√	√	√	√	-	-	√	-	√	√	√	-	-

Tabelle 6.3: Gegenüberstellung anhand multimediaspezifischer Kriterien

In Tabelle 6.3 sind die multimediaspezifischen Kriterien den Anfragesprachen gegenübergestellt. Die erste Spalte der Tabelle enthält die verschiedenen, in Kapitel 5 vorgestellten, Anfragesprachen. In den anderen Spalten wird die Gültigkeit einzelner Kriterien für die jeweilige Anfragesprache dargestellt. Dazu enthält der Kopf der Tabelle zu jedem Kriterium eine Auflistung wesentlicher Eigenschaften der Anfragesprachen, die zur Erfüllung dieses Kriteriums beitragen. Um die Fülle der möglichen Eigenschaften in einer Tabelle unterzubringen, wird eine abkürzende Schreibweise verwendet. Analog zu Tabelle 6.1 und 6.2 gibt ein „√“ die Erfüllung und ein „-“ die Nichterfüllung einer Eigenschaft an. Die einzelnen Buchstaben haben abhängig vom Kriterium folgende Bedeutung:

- *Universell*: V\B\T\A stehen für Video-/Bild-/Text- und Audiodatentypen, die durch die Anfragesprache unterstützt werden.
- *Inhaltsanfragen*: F steht für (f)eaturebasierte Anfragen, S für Anfragen an die (S)truktur der Multimedia-Daten und AV für die Unterstützung (a)bgeschwächter bzw. (v)erfeinernder Anfragen.
- *Raum*: R\T\A\M geben an, ob die Anfragesprache (R)ichtungsbeziehungen, (t)opologische Beziehungen, (A)ggregatfunktionen und (M)engenoperationen auf räumlichen Daten unterstützt.
- *Zeit*: P\O kennzeichnen die Möglichkeiten einer Anfragesprache, zeitbezogene (P)rädikate und (O)perationen für die Anfrageformulierung zu verwenden.

- *unscharfe Anfragen*: durch $P \setminus G \setminus \ddot{A} \setminus R$ wird ausgedrückt, daß die Anfragesprache unscharfe (P)rädikate, (g)ewichtete Anfrageterme, (\ddot{A})hnlichkeitsoperatoren und (R)elevanzwerte anbietet.
- *Präsentation*: die Buchstaben $R \setminus Z \setminus TF \setminus TL \setminus Q$ stehen für eine Unterstützung von (r)äumlichen Präsentationsparametern, (z)eitlichen Präsentationsparametern, (T)rans(f)ormationsfunktionen, (T)rans(l)ationsfunktionen und (Q)uality of Service Parameter durch die Anfragesprache.

Durch die Angabe von $-|\sqrt{|\sqrt{|-}}|$ in der Spalte „Universell“ für die Anfragesprache PICQUERY+ wird somit ausgedrückt, daß dieser Sprachansatz die Medientypen Bild und Text unterstützt, aber keine Möglichkeit für Anfragen an Video- und Audiodaten bietet. Nachfolgend werden die in Tabelle 6.3 dargestellten Zusammenhänge kurz kommentiert.

Nur ein Teil der vorgestellten Ansätze für Anfragesprachen kann aufgrund der Unterstützung sämtlicher gängiger Medientypen als universell bezeichnet werden. Dies sind die Sprachen MMSQL, MOQL sowie die durch die Frameworks SMDS und MADT angebotenen Anfragesprachen. Die restlichen Anfragesprachen unterstützen lediglich einzelne Medientypen, wie zum Beispiel die Anfragesprache des KEQL, welche nur Bild- und Textanfragen anbietet.

Die Möglichkeit der Formulierung inhaltsbasierter Anfragen wird von allen untersuchten Anfragesprachen in irgendeiner Form angeboten. So wird in den meisten Fällen eine Suche anhand von Featurewerten und eine Formulierung von Anfragen an die Struktur der Multimedia-Daten unterstützt. Aber nur einige Anfragesprachen, wie MMSQL, Score und SMDS, ermöglichen verfeinernde Anfragen, um dadurch die Qualität der Anfrageergebnisse zu verbessern.

Räumliche Anfragen können in den meisten Anfragesprachen formuliert werden. Dabei bestehen aber große Unterschiede in den angebotenen Prädikaten und Operationen. Besonders gute Unterstützung räumlicher Anfragen bietet die Sprache MOQL, welche auch räumliche Aggregatfunktionen und Mengenoperationen auf räumlichen Daten bereitstellt. Im Gegensatz dazu können in der Sprache VideoSQL räumliche Anfragen nur formuliert werden, falls vom Benutzer den einzelnen Video-Objekten eine Beschreibung der räumlichen Beziehungen hinzugefügt wurde.

Bis auf Score und CSQL, welche sich auf nicht-zeitliche Medientypen beschränken, werden von allen vorgestellten Anfragesprachen zeitliche Anfragen unterstützt. Für die Formulierung von Anfragen werden dabei größtenteils die 13 Zeitintervallbeziehungen angeboten, wie sie in Abschnitt 3.3 vorgestellt wurden. Unterschiede zwischen den einzelnen Anfragesprachen existieren bei den angebotenen zeitbezogenen Funktionen.

Sehr gravierende Unterschiede gibt es bei der Unterstützung von unscharfen Anfragen durch die einzelnen Anfragesprachen. So bietet VideoSQL hierfür keinerlei Möglichkeiten und in den Anfragesprachen PICQUERY+ und MOQL werden nur Ähnlichkeitsoperatoren unterstützt. Sehr gut erfüllt wird das Kriterium „unscharfe Anfragen“ von der Sprache

MMSQL, welche unscharfe Prädikate, gewichtete Anfrageterme, Ähnlichkeitsoperatoren und ein Relevanzfeedback anbietet.

Ebenfalls sehr unterschiedlich sind die von den Anfragesprachen angebotenen Möglichkeiten bei der Präsentation der Anfrageergebnisse. Diese reichen von keinerlei Unterstützung der Präsentation (MMSQL und SMDS) über die Bereitstellung grundlegender Darstellungsfunktionen (PICQUERY+ und KEQL) bis zur Möglichkeit der Beschreibung von Multimediapräsentationen (MOQL). Keine der untersuchten Anfragesprachen bietet eine Unterstützung der Medientranslation (die Ausgabe eines Medienobjektes in einem anderen Medientyp). Auch QoS-Parameter können in keiner der betrachteten Anfragesprachen angegeben werden.

6.2 Klassifikation von Multimedia-Anfragesprachen

Im folgenden werden einige mögliche Klassifikationen für Anfragesprachen in Multimedia-Datenbanken vorgestellt. Diese basieren auf Beobachtungen bei der Beschreibung der Sprachansätze in Kapitel 5 und dem Vergleich der Anfragesprachen im vorhergehenden Abschnitt.

Eine Möglichkeit für die Klassifizierung der Sprachansätze wird bereits bei der Einteilung des Kapitels 5 in Abschnitte benutzt, indem zwischen Ansätzen, die auf einer Standardanfragesprache beruhen, und neu entwickelten Sprachansätzen unterschieden wird. Die auf der Erweiterung einer Standardanfragesprache basierenden Multimedia-Anfragesprachen können weiter unterschieden werden nach Erweiterungen von SQL und Erweiterungen von OQL.

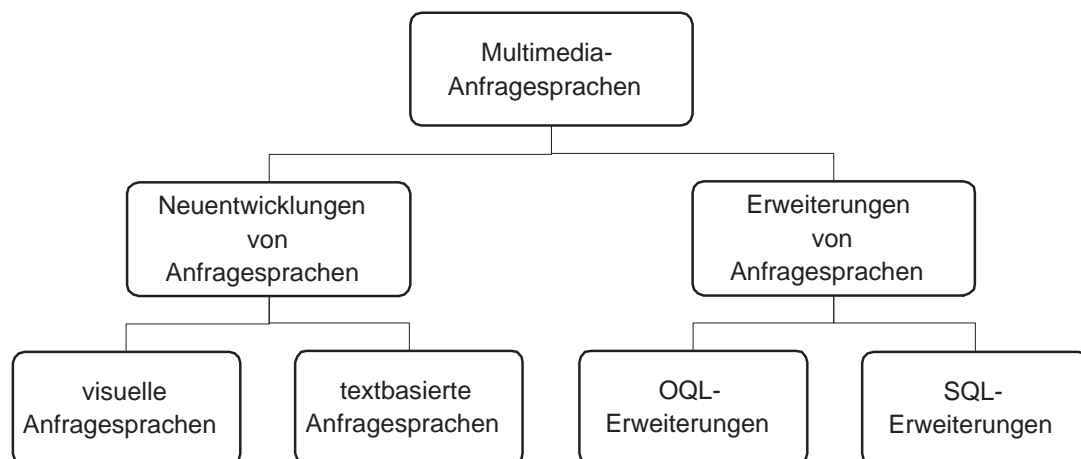


Abbildung 6.1: Klassifikation von Multimedia-Anfragesprachen

Eine Klassifikation ist ebenfalls möglich nach der Art und Weise, wie Anfragen in den einzelnen Sprachen formuliert werden können. Hierbei lässt sich zwischen visuellen Anfragesprachen und textbasierten Anfragesprachen unterscheiden. In Abbildung 6.1 sind beide

Arten der Klassifikation zusammengefaßt dargestellt. Da SQL/OQL-Erweiterungen von Natur aus textbasiert sind, macht eine Unterscheidung nach visuellen und textbasierten Anfragen nur für die neu entwickelten Anfragesprachen einen Sinn.

Die in dieser Arbeit vorgestellten Multimedia-Anfragesprachen können der Klassifikation in Abbildung 6.1 wie folgt zugeordnet werden:

- *visuelle, neu entwickelte Anfragesprachen*: PICQUERY+, SCORE, TVQL,
- *textbasierte, neu entwickelte Anfragesprachen*: SMDS,
- *SQL-Erweiterungen*: KEQL, CSQL, MADT und VideoSQL,
- *OQL-Erweiterungen*: MMSQL, MOQL.

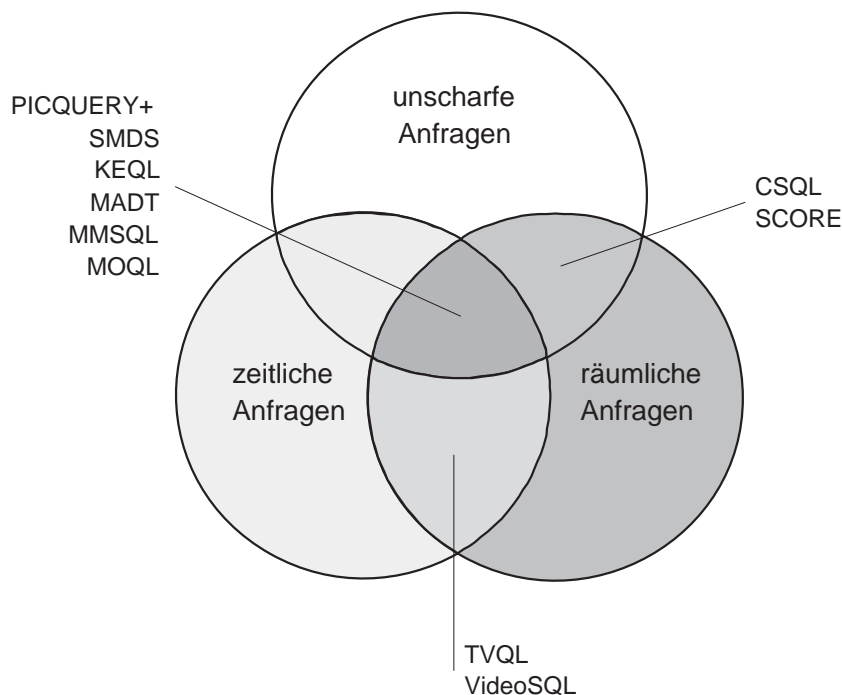


Abbildung 6.2: Klassifikation nach Arten von Anfragen

Anhand der Fähigkeiten, welche die einzelnen untersuchten Anfragesprachen aufweisen, können weitere Klassifikationen vorgenommen werden:

- *Einteilung nach unterstützten Medientypen*: Entsprechend den Medientypen, für die eine Anfragesprache geeignete Konstrukte anbietet, um eine inhaltsbasierte Suche durchführen zu können, läßt sich unterscheiden zwischen:
 1. Bildanfragesprachen: PICQUERY+, SCORE, KEQL, CSQL,
 2. Videoanfragesprachen: TVQL, VideoSQL und

3. universellen Anfragesprachen: SMDS, MADT, MMSQL, MOQL.

- *Einteilung nach Arten von Anfragen*: Anhand der unterstützten Anfragetypen kann eine Klassifikation der Anfragesprachen wie in Abbildung 6.2 vorgenommen werden. Da alle untersuchten Multimedia-Anfragesprachen inhaltsbasierte Anfragen unterstützen, kann dieser Anfragetyp bei der Klassifikation weggelassen werden.

6.3 Bewertung der untersuchten Anfragesprachen

In diesem Abschnitt soll, aufbauend auf den Ergebnissen des Vergleiches der Anfragesprachen, festgestellt werden, inwieweit die vorgeschlagenen Ansätze als Multimedia-Anfragesprache geeignet sind. Eine Anfragesprache muß hierfür nicht alle in Kapitel 4 aufgestellten Kriterien erfüllen. Es gibt aber einige Kriterien, welche ein absolutes Muß in bezug auf die Eignung einer Sprache als Multimedia-Anfragesprache darstellen.

Eine dieser grundlegenden Anforderungen ist die Unterstützung universeller Anfragen, das heißt, medienübergreifende Anfragen an die wichtigsten Medientypen (Video, Bild, Text und Audio) sollten möglich sein. Ein Großteil der vorgestellten Anfragesprachen erfüllt diese Anforderung aber nur sehr unzureichend, wie anhand der Klassifikation nach unterstützten Medientypen im vorhergehenden Abschnitt zu erkennen ist. So sind die Anfragesprachen TVQL und VideoSQL nur für Videodaten verwendbar und Score, PICQUERY+, KEQL und CSQL unterstützen ausschließlich Anfragen an Bild und Textdaten. Die meisten dieser Ansätze lassen sich aufgrund ihrer festen Struktur auch nur schwer erweitern. Eine Integration weiterer Medientypen ist deshalb kaum möglich. Somit sind diese Sprachansätze als universelle Anfragesprachen in MMDBMS nicht verwendbar. Bei den visuellen Anfragesprachen PICQUERY+, TVQL und Score ergibt sich zusätzlich das Problem, daß Anfragen ausschließlich über grafische Benutzeroberflächen spezifiziert werden. Diese Methode ist aber nur für ad-hoc Anfragen geeignet. Eine Einbindung in Anwendungsprogramme ist nicht möglich, wodurch diese Sprachansätze für den Einsatz in MMDBMS ebenfalls ungeeignet sind.

Im Gegensatz zu den zuvor betrachteten Ansätzen können MMSQL und MOQL sowie die Anfragesprachen der beiden Frameworks SMDS und MADT als universell bezeichnet werden. Darüber hinaus bieten die Datenmodelle von MMSQL, MADT und SMDS durch ihre framework-basierte Architektur sehr gute Möglichkeiten zur Erweiterung eines MMDBMS und der Anfragesprache um zusätzliche Medientypen, Features, Prädikate und Operationen. Auch MOQL läßt sich durch das objektorientierte Datenmodell und OQL als Sprachgrundlage erweitern. Die Erweiterbarkeit ist eine wichtige Anforderung für Multimedia-Anfragesprachen, da ein MMDBMS unmöglich alle Features und Operationen standardmäßig enthalten kann, die in den verschiedenen Anwendungen benötigt werden.

Prinzipiell sind die vier universellen Sprachansätze (SMDS, MADT, MMSQL und MOQL) als Multimedia-Anfragesprache geeignet. Jede dieser Sprachen besitzt aber

Anfragesprache	Stärken	Schwächen
SMDS	-theoretisch fundiert -verfeinernde Anfragen	-keine Aggregatfunktionen -keine Präsentationsunterstützung -kein Relevanzfeedback bei unscharfen Anfragen
MADT	-gute Raum-/Zeitunterstützung -ermöglicht unscharfe Anfragen	-kaum Präsentationsunterstützung
MMSQL	-gute Raum-/Zeitunterstützung -verfeinernde Anfragen -umfangreiche unscharfe Anfragemethoden	-keine Präsentationsunterstützung
MOQL	-gute Raum-/Zeitunterstützung -grundlegende Präsentationsunterstützung	-kaum unscharfe Anfragemethoden -keine Relevanzwerte bei Ähnlichkeitsoperationen

Tabelle 6.4: Stärken und Schwächen universeller Multimedia-Anfragesprachen

kleinere Schwächen, die zusammen mit den Stärken in Tabelle 6.4 aufgeführt sind und nachfolgend kurz diskutiert werden.

Die im Zusammenhang mit dem Framework für eine strukturierte Multimedia-Datenbank (SMDS) vorgestellte logikbasierte Anfragesprache besitzt durch deren formale Definition eine gute theoretische Basis. Außerdem wird eine Verfeinerung bzw. Verschärfung von Anfragen unterstützt. Leider werden keine Methoden für die Präsentation der Anfrageergebnisse angeboten und keine Relevanzwerte für unscharfe Anfragen unterstützt.

Medienspezifische abstrakte Datentypen (MADT) mit SQL3 als Anfragesprache bieten durch eine Reihe von Prädikaten und Operationen gute Möglichkeiten für räumliche und zeitbezogene Anfragen. Des Weiteren ist mit Hilfe von gewichteten Anfragetermen und Ähnlichkeitsoperationen mit Relevanzwerten eine Formulierung unscharfer Anfragen möglich. Ein Schwachpunkt der MADT's ist die fehlende Unterstützung einer Ergebnispräsentation für Multimedia-Datentypen. Prinzipiell ließe sich SQL3 hierfür um geeignete Konstrukte erweitern, wenn die MADT's die nötige Funktionalität bereitstellen würden.

Die Stärke von MMSQL sind unscharfe Anfragen. Hierfür werden alle vorgestellten Konzepte, wie unscharfe Prädikate, gewichtete Anfrageterme, Ähnlichkeitsoperatoren und -anfragen sowie verfeinernde Anfragen, im Sprachentwurf berücksichtigt. Es mangelt MMSQL aber wie den schon zuvor betrachteten Sprachen, an Konstrukten zur Präsentation der Anfrageergebnisse, obwohl dies vom Datenmodell unterstützt wird.

Die einzige der vier universellen Anfragesprachen, welche Methoden für die Ergebnispräsentation anbietet, ist MOQL. Mit Hilfe einer **present**-Klausel lassen sich einfa-

che Präsentationen und Darstellungsfunktionen für Multimedia-Datentypen durch die Anfragesprache spezifizieren. Außerdem stellt MOQL zahlreiche Konstrukte für räumliche und zeitliche Anfragen zur Verfügung. Für unscharfe Anfragen wird aber lediglich ein Prädikat für Ähnlichkeitsvergleiche ohne Relevanzwerte angeboten, was ein klarer Schwachpunkt dieser Sprache ist.

Zusammenfassend läßt sich feststellen, daß ein Großteil der untersuchten Anfragesprachen aufgrund fehlender Unterstützung grundlegender Medientypen und mangelnder Erweiterbarkeit für den Einsatz in Multimedia-Datenbanken ungeeignet sind. Sie haben aber als Anfragesprachen in speziellen Video- und Bilddatenbanksystemen durchaus ihre Berechtigung. Die zuvor als universell klassifizierten Ansätze MMSQL, MOQL, MADT und SMDS sind grundsätzlich als Multimedia-Anfragesprachen geeignet, enthalten aber einige Mängel. Die größten Probleme treten dabei im Zusammenhang mit unscharfen Anfragen und der Ergebnispräsentation auf. So bietet keine der untersuchten Anfragesprachen Konstrukte für eine Medientranslation oder zur Spezifikation von QoS-Parametern an.

6.4 Entwurfsprinzipien für Multimedia-Anfragesprachen

Abschließend sollen in diesem Kapitel Entwurfsprinzipien für zukünftige Entwicklungen von Multimedia-Anfragesprachen aufgestellt werden. Diese ergeben sich aus den multimediaspezifischen Kriterien in Abschnitt 4.2 sowie als Schlußfolgerung aus dem Vergleich der bisherigen Ansätze für Multimedia-Anfragesprachen. Im einzelnen sollte beim Entwurf folgendes beachtet werden:

1. *Universell*: Eine Anfragesprache sollte universell sein und somit alle grundlegenden Multimedia-Datentypen unterstützen. Außerdem muß eine Erweiterung um zusätzliche Datentypen möglich sein, wobei sich diese leicht in die Anfragesprache integrieren lassen sollten. Ein framework-basierter Ansatz, wie im Zusammenhang mit den MADT's vorgestellt, ist hierfür gut geeignet.
2. *Inhaltsbasierte Anfragen*: Für eine inhaltsbasierte Suche müssen Anfragen an Metadaten formuliert werden können. Einerseits werden hierfür Anfragekonstrukte an inhaltsbeschreibende Attribute mit einer Abschwächung/Verschärfung von Anfrageattributen anhand von Begriffshierarchien benötigt. Andererseits müssen sich die Indexstrukturen, welche mit Hilfe der aus den Medienobjekten abgeleiteten Features vom MMDDBMS erstellt werden, in Anfragen verwenden lassen .
3. *Räumliche und zeitliche Beziehungen*: Zur Formulierung von Anfragen müssen räumliche und zeitliche Beziehungen zwischen den Medienobjekten verwendet werden können. Im wesentlichen sollten hierzu topologische, Richtungs- und Distanzbeziehungen für räumliche Daten sowie Zeitintervallbeziehungen für zeitbezogene

Daten durch Prädikate unterstützt werden. Des weiteren benötigt man Aggregatfunktionen und Mengenoperationen auf räumlichen und zeitlichen Daten.

4. *Unschärfe Anfrageprädikate*: Eine Multimedia-Anfragesprache sollte unscharfe Anfrageprädikate enthalten. Da solche Prädikate vom Anwendungsgebiet abhängig sind, müssen diese durch den Anwendungsentwickler manipuliert werden können. Ein framework-basierter Ansatz bietet hierfür wiederum gute Voraussetzungen.
5. *Wichtung*: Ebenfalls sollte zur Unterstützung unscharfer Anfragen einzelnen Anfragebedingungen entsprechend der Priorität in bezug auf die gesamte Anfrage eine Wichtung zugeordnet werden können.
6. *Ähnlichkeitsanfragen*: Da nicht immer eine exakte Anfrageformulierung möglich ist, muß eine Multimedia-Anfragesprache Ähnlichkeitsoperatoren anbieten, um so mit Hilfe von Beispielobjekten nach anderen Medienobjekten zu suchen. Da hierbei ein Ähnlichkeitsvergleich anhand verschiedener Merkmale möglich ist, werden mehrere Ähnlichkeitsoperationen benötigt, wobei die Syntax für alle Operatoren einheitlich sein sollte. Den Ergebnisobjekten einer Ähnlichkeitsanfrage muß ein Relevanzwert zugeordnet werden.
7. *Präsentation*: Eine Multimedia-Anfragesprache muß auch die Präsentation der Anfrageergebnisse durch geeignete Konstrukte unterstützen. Hierfür müssen Operationen zur Medientransformation und -translation sowie QoS-Parameter durch die Anfrage spezifiziert werden können. Eine Beschreibung kompletter Präsentationen sollte nicht Aufgabe der Anfragesprache sein. Dies kann vielmehr durch geeignete Präsentationssprachen oder grafische Benutzeroberflächen realisiert werden.
8. *Benutzerschnittstelle*: Für die ad-hoc-Formulierung von Anfragen wird ein Benutzerinterface benötigt. Dieses sollte folgende Möglichkeiten anbieten:
 - eine Anfrageformulierung mit Hilfe grafischer Methoden, zum Beispiel für die Erstellung von Bildskizzen für eine Ähnlichkeitssuche in Bilddaten,
 - Unterstützung eines iterativen Anfrageprozesses und
 - räumliche und zeitliche Präsentation der Anfrageergebnisse mit Interaktionsmöglichkeiten für den Benutzer (z.B. Start, Stop, Vor- und Rücklauf für Videosequenzen)

Zusätzlich zu den oben aufgeführten Punkten sollten beim Entwurf einer Multimedia-Anfragesprache auch Anforderungen beachtet werden, die allgemein für jede Anfragesprache gültig sind. Solche Anforderungen sind bereits in Abschnitt 4.1 festgelegt worden und sollen an dieser Stelle nicht noch einmal wiederholt werden.

7 | Zusammenfassung und Ausblick

In diesem abschließenden Kapitel werden die Ergebnisse der Arbeit zusammengefaßt und kommentiert. Außerdem wird ein Ausblick auf mögliche weitere Forschungsaktivitäten im Zusammenhang mit Multimedia-Anfragesprachen gegeben.

7.1 Ergebnisse

Ziel dieser Arbeit war es, einen Überblick über aktuelle Ansätze für Anfragesprachen in Multimedia-Datenbanken zu erstellen. Außerdem sollten die einzelnen Sprachansätze analysiert und anhand von ausgewählten Kriterien miteinander verglichen werden, um dadurch die Leistungsfähigkeit, aber auch die Defizite aktueller Multimedia-Anfragesprachen aufzuzeigen.

Dazu wurden als Grundlage für weitere Untersuchungen zu Beginn dieser Arbeit wichtige Begriffe festgelegt sowie die Merkmale von Multimedia-Datenbank-Management-Systemen beschrieben. Besondere Beachtung fanden dabei die verschiedenen Medientypen und deren charakteristische Eigenschaften. In diesem Zusammenhang wurde auch die Verwendung von Metadaten als eine Möglichkeit zur Beschreibung des komplexen Inhaltes von Multimedia-Daten erläutert. Außerdem erfolgte eine kurze Einführung in die Anfragesprachen SQL und OQL, da diese als Basis für viele der im weiteren Verlauf der Arbeit vorgestellten Multimedia-Anfragesprachen dienen.

Aufgrund der speziellen Eigenschaften von Multimedia-Daten im Vergleich zu alphanumerischen Daten reichen die Möglichkeiten zur Anfrageformulierung, wie sie von traditionellen DBMS bekannt sind, nicht aus. Zusätzlich werden in einem MMDBMS weitere Konzepte zur Formulierung von Anfragen benötigt. Als Ergebnis eines umfangreichen Literaturstudiums wurden die Konzepte zur Formulierung von Multimedia-Anfragen nach inhaltsbasierten, räumlichen, zeitlichen sowie unscharfen Anfragen klassifiziert. Dabei wurden auch verschiedene Methoden zur Unterstützung der einzelnen Arten von Anfragen untersucht. Als eine weitere wichtige Aufgabe von Multimedia-Anfragesprachen stellte sich in diesem Zusammenhang die Unterstützung der Präsentation der Anfrageergebnisse heraus. Im Gegensatz zu Anfragen an traditionelle Datenbank-Management-Systeme können die Anfrageergebnisse nicht einfach als Tabellen ausgegeben werden. Es werden vielmehr spezielle Darstellungsmethoden benötigt, welche durch die Anfragesprache zu spezifizieren sind.

Die Grundlage für einen ordentlichen Vergleich von Multimedia-Anfragesprachen sind geeignete Kriterien, anhand derer sich die Leistungsfähigkeit einzelner Sprachansätze messen läßt. In dieser Arbeit wurde zwischen allgemeinen Kriterien und multimedia-spezifischen Kriterien unterschieden. Die allgemeinen Kriterien enthalten Anforderungen, welche für jede Art von Anfragesprache gültig sind. Basierend auf verschiedenen Veröffentlichungen wurde eine umfangreiche Liste von allgemeinen Kriterien aufgestellt. Multimediaspezifische Kriterien sind Anforderungen, die sich aufgrund der speziellen Eigenschaften von Multimedia-Daten ergeben. Da bisher noch keine multimediaspezifischen Kriterien für Anfragesprachen in der Literatur veröffentlicht wurden, mußten diese im Rahmen der Arbeit aufgestellt werden. Grundlage hierfür war die Untersuchung der verschiedenen Arten von Anfragen in MMDBMS in Kapitel 3.

Einen Überblick über aktuelle Forschungsansätze für Multimedia-Anfragesprachen zu geben, war eines der Ziele dieser Arbeit. Die verschiedenen Sprachansätze wurden dazu in die drei Kategorien – neu entwickelte, SQL-basierte und OQL-basierte Anfragesprachen – eingeteilt. Basierend auf einer umfangreichen Literaturrecherche wurden zu jeder der Kategorien eine Reihe von Anfragesprachen vorgestellt. Aufgrund des zeitlichen Rahmens dieser Arbeit beschränkt sich die Beschreibung der einzelnen Sprachansätze auf die Erläuterung grundlegender Konstrukte und Merkmale der Sprachen. In Anbetracht zahlreicher Forschungsaktivitäten auf dem Gebiet der Multimedia-Datenbanken kann leider kein Anspruch auf Vollständigkeit erhoben werden.

Aufbauend auf der Beschreibung der einzelnen Anfragesprachen und den aufgestellten Kriterien wurden die Sprachansätze analysiert und miteinander verglichen. Dabei konnten teilweise sehr große Unterschiede zwischen den einzelnen Sprachen in bezug auf ihre Eignung als Multimedia-Anfragesprache festgestellt werden. Viele der untersuchten Sprachansätze unterstützen nur einzelne Medientypen und sind deshalb für die Verwendung in MMDBMS ungeeignet. Lediglich vier der Anfragesprachen können als universell bezeichnet werden, da sie eine Unterstützung aller Grundmedientypen anbieten. Aber auch diese universellen Anfragesprachen weisen verschiedene Mängel auf, zum Beispiel bei den Möglichkeiten zur Spezifikation der Ergebnispräsentation oder im Zusammenhang mit unscharfen Anfragen. Sehr gut erfüllen die meisten der untersuchten Anfragesprachen die Anforderungen für eine räumliche und zeitbezogene Suche. Basierend auf den Ergebnissen des Vergleiches wurden die Sprachansätze klassifiziert und abschließend Entwurfsprinzipien für die Entwicklung zukünftiger Multimedia-Anfragesprachen aufgestellt.

7.2 Ausblick

Die Ergebnisse dieser Arbeit zeigen deutlich, daß eine weitere Forschung auf dem Gebiet der Multimedia-Anfragesprachen nötig ist. Dabei könnte unter anderem untersucht werden, ob sich SQL3, welche als Anfragesprache für medienspezifische abstrakte Datentypen vorgestellt wurde, prinzipiell zu einer kompletten Multimedia-Anfragesprache erweitern läßt. Interessant ist dies vor allem deshalb, weil sich dadurch eine Standard-

Multimedia-Anfragesprache entwickeln ließe.

Des weiteren muß grundsätzlich die Frage geklärt werden, inwieweit die Spezifikation der Präsentation von Anfrageergebnissen Aufgabe der Anfragesprache ist. In einigen Sprachansätzen wird versucht, eine komplette Multimediapräsentation zusammen mit den Suchbedingungen durch die Anfragesprache zu beschreiben. Sinnvoller wäre eventuell eine Zweiteilung des Anfrageprozesses, bei dem zunächst die gewünschten Daten ausgewählt werden und anschließend eine Beschreibung der Präsentation dieser Daten mit Hilfe von Präsentationssprachen erfolgt.

Genereller Forschungsbedarf besteht auch bei den theoretischen Grundlagen von Multimedia-Anfragesprachen. Erste Schritte in diese Richtung wurden lediglich von Marcus et al. durch die formale Definition einer strukturierten Multimedia-Datenbank und einer logikbasierten Anfragesprache unternommen. Für keine der anderen Sprachansätze wurde in den entsprechenden Veröffentlichungen eine formale Semantik angegeben.

Zusammenfassend läßt sich feststellen, daß ein Teil der aktuellen Forschungsansätze für Multimedia-Anfragesprachen bereits grundlegende Anforderungen für eine Suche in Multimedia-Datenbanken erfüllen. Die einzelnen Sprachansätze weisen aber eine Reihe von Defiziten auf. Um diese zu beseitigen, sind zukünftig weitere Forschungsaktivitäten notwendig.

Literaturverzeichnis

- [ABH97] Apers, P. M. G.; Blanken, H.; Houtsma, M. A. W. (Hrsg.): *Multimedia Databases in Perspective*. Springer Verlag, London, 1997.
- [All83] Allen, J. F.: Maintaining knowledge about temporal intervals. *Communications of ACM*, Band 26, Nr. 11, S. 832–843, 1983.
- [AMS98] Amato, G.; Mainetto, G.; Savino, P.: An Approach to a Content-Based Retrieval of Multimedia Data. In Subrahmanian, V. S.; Tripathi, S. K. (Hrsg.): *Multimedia Information Systems*, S. 9–36. Kluwer Academic Publishers, 1998.
- [AN97] Adjero, D. A.; Nwosu, K.: Multimedia Database Management - Requirements and Issues. *IEEE MultiMedia*, Band 4, Nr. 3, S. 24–33, Juli 1997.
- [ATN99] Aberer, K.; Thimm, H.; Neuold, E.: Multimedia Database Management Systems. In Furht, B. (Hrsg.): *Handbook of Multimedia Computing*, S. 285–308. CRC Press LLC, Boca Raton, 1999.
- [ATY⁺95] Aslandoglu, Y. A.; Thier, C.; Yu, C. T.; Lin, C.; Nair, K. R.: Design, Implementation and Evaluation of SCORE (a System for Content based Retrieval of Pictures). In *Proceedings of the 11th International Conference on Data Engineering*, IEEE ICDE, S. 280–287. IEEE Computer Society, Taipei, Taiwan, März 1995.
- [BCF99] Bertino, E.; Catania, B.; Ferrari, E.: Multimedia IR: Models and Languages. In Baeza-Yates, R.; Ribeiro-Neto, B. (Hrsg.): *Modern Information Retrieval*, S. 325–343. ACM Press, Essex, England, 1999.
- [BGN98] Baral, C.; Gonzalez, G.; Nandigam, A.: SQL+D: extended display capabilities for multimedia database queries. In *Proceedings ACM Multimedia 98*, ACM Multimedia, S. 109–114. ACM Press, Bristol, England, September 1998.
- [BGS98] Baral, C.; Gonzalez, G.; Son, T. C.: Design and implementation of display specifications for multimedia answers. In *Proceedings of the 14th International Conference on Data Engineering*, S. 558–565. IEEE Computer Society, Orlando, Florida, Februar 1998.

- [BKS98] Boll, S.; Klas, W.; Sheth, A.: Overview on Using Metadata to Manage Multimedia Data. In Sheth, A.; Klas, W. (Hrsg.): *Multimedia Data Management Using Metadata to Integrate and Apply Digital Media*, S. 1–24. McGraw-Hill, New York, 1998.
- [BR99] Baeza-Yates, R.; Ribeiro-Neto, B.: *Modern Information Retrieval*. ACM Press, Essex, England, 1999.
- [Cat94] Cattell, R. G. G. (Hrsg.): *The Object Database Standard: ODMG-93*. Morgan Kaufmann Publishers, San Mateo, CA, 1994.
- [CHIT98] Chu, W. W.; Hsu, C. C.; Jeong, I. T.; Taira, R. K.: Content-Based Image Retrieval Using Metadata and Relaxation Techniques. In Sheth, A.; Klas, W. (Hrsg.): *Multimedia Data Management Using Metadata to Integrate and Apply Digital Media*, S. 149–187. McGraw-Hill, New York, 1998.
- [CIT⁺93] Cardenas, A. F.; Jeong, I. T.; Taira, R. K.; Barker, R.; Breant, C. M.: The Knowledge-Based Object-Oriented PICQUERY+ Language. *IEEE Transaction on Knowledge and Data Engineering*, Band 5, Nr. 4, S. 644–657, August 1993.
- [DAPG98] Day, Y. F.; Al-Khatib, W.; Paul, R.; Ghafoor, A.: Specification of a Query Language for Multimedia Database Systems. In Hirakawa, M.; Bhattacharya, S. (Hrsg.): *International Workshop on Multimedia Software Engineering*, S. 111–119. IEEE Computer Society, Los Alamitos, California, April 1998.
- [DC98] Dionisio, J. D. N.; Cardenas, A. F.: Advances in Image Database Languages. In Wong, S. T. C. (Hrsg.): *Medical Image Databases*, S. 173–199. Kluwer Academic Publishers, Norwell, USA, 1998.
- [DD97] Date, C. J.; Darwen, H.: *A Guide to the SQL Standard*. Addison-Wesley, Reading, MA, 4. Auflage, 1997.
- [DSS98] Dalamagas, T.; Sellis, T.; Sinos, L.: A Visual Database System for Spatial and Non-spatial Data Management. In Ioannidis, Y.; Klas, W. (Hrsg.): *Visual Database Systems 4*, S. 105–122. Chapman and Hall, Gateshead, Great Britain, 1998.
- [EF91] Egenhofer, M.; Franzosa, R.: Point-set topological spatial relations. *International Journal of Geographical Information Systems*, Band 5, Nr. 2, S. 161–174, 1991.
- [Fag98] Fagin, R.: Fuzzy Queries in Multimedia Database Systems. Research Report RJ 10106, IBM Research Division, San Jose, California, März 1998.

- [Fal99] Faloutsos, C.: Multimedia IR: Indexing and Searching. In Baeza-Yates, R.; Ribeiro-Neto, B. (Hrsg.): *Modern Information Retrieval*, S. 345–365. ACM Press, Essex, England, 1999.
- [FSN⁺97] Flickner, M.; Sawhney, H.; Niblack, W.; Ashley, J.; Huang, Q.; Dom, B.; Gorkani, M.; Hafner, J.; Lee, D.; Petkovic, D.; Steele, D.; Yanker, P.: Query by Image and Video Content: The QBIC System. In Maybury, M. T. (Hrsg.): *Intelligent Multimedia Information Retrieval*, S. 7–22. AAAI Press / The MIT Press, Menlo Park, CA, 1997.
- [Heu97] Heuer, A.: *Objektorientierte Datenbanken: Konzepte, Modelle, Standards und Systeme*. Addison-Wesley, Bonn, 2. Auflage, 1997.
- [Hje95] Hjelsvold, R.: *VideoSTAR - A Database for Video Information Sharing*. PhD thesis, Department of Computer Systems and Telematics, Norwegian Institute of Technology, November 1995.
- [HK96] Hirzalla, N.; Karmouch, A.: A Multimedia Query Spezifikation Language. In Nwosu, K. C.; Thuraisingham, B. M.; Berra, P. B. (Hrsg.): *Multimedia Database Systems: Design and Implementation Strategies*, S. 160–183. Kluwer Academic Publishers, Norwell, Massachusetts, 1996.
- [HR96] Hibino, S.; Rundensteiner, E. A.: A Visual Multimedia Query Language For Temporal Analysis of Video Data. In Nwosu, K. C.; Thuraisingham, B. M.; Berra, P. B. (Hrsg.): *Multimedia Database Systems: Design and Implementation Strategies*, S. 123–159. Kluwer Academic Publishers, Norwell, Massachusetts, 1996.
- [HS97] Heuer, A.; Saake, G.: *Datenbanken — Konzepte und Sprachen, 1. korrigierter Nachdruck*. International Thomson Publishing, Bonn, 1997.
- [HSH⁺98] Hollfelder, S.; Schmidt, F.; Hemmje, M.; Aberer, K.; Steinmetz, A.: Transparent Integration of Continuous Media Support into a Multimedia DBMS. In *Proceedings of the International Workshop on Issues and Applications of Database Technology*. Society for Design and Process Science, Berlin, Juli 1998.
- [Inf97a] Informix Press: *Excalibur Image DataBlade Module, User's Guide Version 1.1*, Juli 1997.
- [Inf97b] Informix Press: *Excalibur Text Search DataBlade Module, User's Guide Version 1.1*, Juli 1997.
- [ISO86] ISO (International Organization for Standardization), ISO 8879–1986: *Information Processing - Text and Office Systems - Standardized Generalized Markup Language (SGML)*, 1986.

- [ISO95a] ISO (International Organization for Standardization) & ANSI (American National Standards Institute), ISO/IEC JTC 1/SC 21/SQL/MM LHR-003, N9789: *ISO-ANSI Working Draft: SQL Multimedia and Application Packages (SQL/MM) Part1: Framework*, September 1995.
- [ISO95b] ISO (International Organization for Standardization) & ANSI (American National Standards Institute), ISO/IEC JTC 1/SC 21/SQL/M LHR-004, N9790: *ISO-ANSI Working Draft: SQL Multimedia and Application Packages (SQL/MM) Part2: Full-Text*, September 1995.
- [ISO95c] ISO (International Organization for Standardization) & ANSI (American National Standards Institute), ISO/IEC JTC 1/SC 21/SQL/MM LHR-006, N9792: *ISO-ANSI Working Draft: SQL Multimedia and Application Packages (SQL/MM) Part4: General Purpose Facilities*, September 1995.
- [ISO96] ISO (International Organization for Standardization) & ANSI (American National Standards Institute), ISO/IEC JTC 1/SC 21/SQL/MM MAD-007: *ISO-ANSI Working Draft: SQL Multimedia and Application Packages (SQL/MM) Part5: Still Image*, Juni 1996.
- [ISO97] ISO (International Organization for Standardization) & ANSI (American National Standards Institute), ISO/IEC JTC 1/SC 21N/CD 13249-3:199x(E): *ISO-ANSI Working Draft: SQL Multimedia and Application Packages (SQL/MM) Part3: Spatial*, April 1997.
- [KB95] Khoshafian, S.; Baker, A. B.: *Multimedia and Imaging Databases*. Morgan Kaufmann Publishers, San Francisco, CA, 1995.
- [KR98] Kaushik, S.; Rundensteiner, E.: Direct-Manipulation Spatial Exploration Using SVIQUEL. In Ioannidis, Y.; Klas, W. (Hrsg.): *Visual Database Systems 4*, S. 179–185. Chapman and Hall, Gateshead, Great Britain, 1998.
- [LC98] Li, W.-S.; Candan, K. S.: SEMCOG: A Hybrid Object-based Image Database System and Its Modeling, Language, and Query Processing. In *Proceedings of the 14th International Conference on Data Engineering*, S. 284–291. IEEE Computer Society, Orlando, Florida, Februar 1998.
- [Li98] Li, J. Z.: *Modeling and Querying Multimedia Data*. Technical Report TR-98-05, Laboratory for Database Systems Research, University of Alberta, März 1998.
- [LÖS96] Li, J. Z.; Özsu, M. T.; Szafron, D.: Spatial Reasoning Rules in Multimedia Management System. In *Proceedings of International Conference on Multimedia Modelling*, S. 119–133. Toulouse, France, November 1996.

- [LÖS97] Li, J. Z.; Özsu, M. T.; Szafron, D.: Multimedia Extensions To Database Query Languages. Technical Report TR 97-01, Laboratory for Database Systems Research, University of Alberta, Edmonton, Alberta, Canada, Januar 1997.
- [LÖSO97] Li, J. Z.; Özsu, M. T.; Szafron, D.; Oria, V.: MOQL: A Multimedia Object Query Language. In *Proceedings of the 3rd International Workshop on Multimedia Information Systems*, S. 19–28. Como, Italy, September 1997.
- [LSB⁺99] Lee, T.; Sheng, L.; Bozkaya, T.; Balkir, N. H.; Özsoyoglu, Z. M.; Özsoyoglu, G.: Querying Multimedia Presentations Based on Content. *IEEE Transaction on Knowledge and Data Engineering*, Band 11, Nr. 3, S. 361–385, Juni 1999.
- [LV97] Lausen, G.; Vossen, G.: *Models and Languages of Object-Oriented Databases*. Addison-Wesley, Harlow, UK, 1997.
- [Mar96] Marcus, S.: Querying Multimedia Databases in SQL. In Subrahmanian, V. S.; Jajodia, S. (Hrsg.): *Multimedia Database System: Issues and Research Direction*, S. 263–276. Springer-Verlag, Berlin, 1996.
- [Mar99] Marder, U.: Medienspezifische Datentypen für objekt-relationale DBMS: Abstraktionen und Konzepte. In Buchmann, A. P. (Hrsg.): *Proc. GI-Fachtagung "Datenbanksysteme in Büro, Technik und Wissenschaft" (BTW'99), Freiburg im Breisgau*, Informatik aktuell, S. 210–231. Springer-Verlag, Berlin, März 1999.
- [MHSA98] Malsy, M.; Hollfelder, S.; Steinmetz, A.; Aberer, K.: Unterstützung des strukturierten Zugriffs auf MPEG Videos in einem Multimedia Datenbankmanagementsystem. In *Proc. of GI-Workshop "Inhaltsbezogene Suche von Bildern und Videosequenzen in digitalen multimedialen Archiven" (KI'98), Bremen, September 1998*, Bericht 6/98, S. 27–34. Bremen, September 1998.
- [MS96] Marcus, S.; Subrahmanian, V. S.: Foundations of Multimedia Information Systems. *Journal of the ACM*, Band 43, Nr. 3, S. 474–523, Mai 1996.
- [MW91] Meyer-Wegener, K.: *Multimedia-Datenbanken*. Leitfäden zur angewandten Informatik. Teubner-Verlag, Stuttgart, 1991.
- [Myr97] Myrach, T.: TSQL2: Der Konsens über eine temporale Datenbanksprache. *Informatik-Spektrum*, Band 20, Nr. 3, S. 143–150, Juni 1997.
- [Nah99] Nahrstedt, K.: Quality of Service in Networked Multimedia Systems. In Furht, B. (Hrsg.): *Handbook of Multimedia Computing*, S. 839–874. CRC Press LLC, Boca Raton, 1999.

- [OÖSI99] Oria, V.; Özsu, M. T.; Szafron, D.; Iglinski, P. J.: Defining Views in an Image Database System. In *8th IFIP 2.6 Working Conference on Database Semantics - Semantic Issues in Multimedia Systems*, S. 231–250. Rotorua, New Zealand, Januar 1999.
- [OT93] Oomoto, E.; Tanaka, K.: OVID: Design and Implementation of a Video-Object Database System. *IEEE Transaction on Knowledge and Data Engineering*, Band 5, Nr. 4, S. 629–643, August 1993.
- [OXÖ99] Oria, V.; Xu, B.; Özsu, M. T.: VisualMOQL: A Visual Query Language for Image Databases. In *Proceedings of the 6th IEEE International Conference on Multimedia Computing and Systems*. Florence, Italy, Juni 1999.
- [PE88] Pullar, D.; Egenhofer, M.: Toward formal definitions of topological relations among spatial objects. In *Proceedings of the 3rd International Symposium on Spatial Data Handling*, S. 165–176. Sydney, Australia, 1988.
- [Pra97a] Prabhakaran, B.: *Multimedia Database Management Systems*. Kluwer Academic Publishers, AH Dordrecht NL, 1997.
- [Pra97b] Prabhakaran, B.: Querying Multimedia Databases. In *Multimedia Database Management Systems*, S. 141–152. Kluwer Academic Publishers, AH Dordrecht NL, 1997.
- [PS95] Papadias, D.; Sellis, T.: A Pictorial Query-By-Example Language. *Journal on Visual Languages and Computing*, Band 6, Nr. 1, S. 53–72, März 1995.
- [SA95] Samet, H.; Aref, W.: Spatial Data Models and Query Processing. In Kim, W. (Hrsg.): *Modern Database Systems*, S. 338–360. ACM Press, New York, NJ, 1995.
- [SC97] Smith, J. R.; Chang, S.: Querying by Color Regions Using the VisualSEEK Content-Based Visual Query System. In Maybury, M. T. (Hrsg.): *Intelligent Multimedia Information Retrieval*, S. 23–41. AAAI Press / The MIT Press, Menlo Park, CA, 1997.
- [Sch97] Schäuble, P.: *Speech Retrieval*. Kluwer Academic Publishers, Boston, 1997.
- [SH99] Saake, G.; Heuer, A.: *Datenbanken — Implementierungstechniken*. MITP-Verlag, Bonn, 1999.
- [Sha99] Sharda, N.: Multimedia Networks. In Furht, B. (Hrsg.): *Handbook of Multimedia Computing*, S. 783–814. CRC Press LLC, Boca Raton, 1999.
- [SST97] Saake, G.; Schmitt, I.; Türker, C.: *Objektdatenbanken — Konzepte, Sprachen, Architekturen*. International Thomson Publishing, Bonn, 1997.

-
- [Sub98] Subrahmanian, V. S.: *Principles of Multimedia Database Systems*. Morgan Kaufmann Publishers, San Francisco, CA, 1998.
- [YM98] Yu, C. T.; Meng, W. (Hrsg.): *Principles of Database Query Processing for Advanced Applications*. Morgan Kaufmann Publishers, San Francisco, CA, 1998.
- [ZCF⁺97] Zaniolo, C.; Ceri, S.; Faloutsos, C.; Snodgrass, R. T.; Subrahmanian, V. S.; Zicari, R.: Overview of Temporal Databases. In *Advanced Database Systems*, S. 99–125. Morgan Kaufmann Publishers, San Francisco, CA, 1997.

Selbständigkeitserklärung

Hiermit erkläre ich, daß ich die vorliegende Arbeit selbständig und nur mit erlaubten Hilfsmitteln angefertigt habe.

Magdeburg, den 31. August 1999

Marco Paskamp

Thesen

1. Die Verwendung von Metadaten ist ein geeignetes Mittel zur Beschreibung des Inhaltes, der Struktur und der Semantik von Multimedia-Daten, welche im allgemeinen ein sehr großes Datenvolumen aufweisen und in ihrer ursprünglichen Form oft als unformatierte Rohdaten vorliegen.
2. Aufgrund der besonderen Merkmale von Multimedia-Daten reichen die Konzepte klassischer Anfragesprachen, wie SQL, zur Formulierung von Anfragen in Multimedia-Datenbanken nicht aus. Für eine effiziente Suche in einem MMDBMS basierend auf dem Inhalt der Multimedia-Daten benötigt man vielmehr eine Multimedia-Anfragesprache, welche neben der Ausdrucksfähigkeit von SQL zusätzliche multimedia-spezifische Erweiterungen enthält.
3. Eine grundlegende Anforderung an eine Multimedia-Anfragesprache ist die Unterstützung aller wichtigen Grundmedientypen und die beliebige Erweiterbarkeit um zusätzliche Mediendatentypen, damit ein breites Spektrum an Anwendungen abgedeckt werden kann.
4. Eine leistungsfähige Multimedia-Anfragesprache muß räumliche und zeitliche Beziehungen, abgeleitet aus den Multimedia-Daten, sowohl zur Formulierung von Anfragen als auch zur Beschreibung der Ergebnispräsentation unterstützen. Um zufriedenstellende Anfragen an ein MMDBMS basierend auf dem Inhalt der Multimedia-Daten stellen zu können, benötigt eine Anfragesprache Konzepte zur unscharfen Suche und zur inhaltsbasierten Suche anhand von Metadaten.
5. Für eine ad-hoc-Formulierung von Anfragen in MMDBMS wird zusätzlich zu einer leistungsfähigen Multimedia-Anfragesprache eine Benutzerschnittstelle benötigt, welches unter anderem die Formulierung komplexer Anfrageprädikate mittels grafischer Methoden, einen iterativen Anfrageprozeß und die Präsentation der Anfrageergebnisse unterstützt.
6. Heutige Multimedia-Anfragesprachen können die an sie gestellten Anforderungen nur bedingt erfüllen, wobei einige der vorhandenen Ansätze den Anforderungen schon sehr nahe kommen. Zukünftig müssen vor allem Konzepte zur Unterstützung der Präsentation von Anfrageergebnissen und für die Formulierung unscharfer Anfragen stärker in die Anfragesprachen integriert werden.