

Broadcast Multimediaübertragung auf der Basis von MPEG-4

Dr.-Ing. Thomas Bonse¹, Dipl.-Ing. Michael Stepping²

Kurzfassung

Der Standard *MPEG-4* legt Mechanismen für die Verknüpfung von Multimediaobjekten fest. Darüber hinaus stellt es ein Konzept für die Übertragung von *MPEG-4* Datenströmen, dem *Delivery Multimedia Integration Framework – DMIF* zur Verfügung.

Dieser Beitrag stellt eine DMIF Implementierung für das Broadcast Szenarium vor. Es wird hierbei insbesondere eine Lösung für die DMIF Signalisierung und für das Umschalten auf den Broadcast Datenstrom zu beliebigen Zeitpunkten (*Late Tuning-In*) aufgezeigt. Zum Einsatz kommt die MPEG-4 Referenzsoftware als Beweis der DMIF Konformität. Die Arbeiten wurden im Rahmen des EU-Projektes „*MPEG-4 PC – MPEG-4 System Implementation and Tools for Personal Computers*“ durchgeführt [1].

1. Einführung

Mit dem Projekt „*Virtuelle Universität - ET-Online*“ [2] hat der Fachbereich Elektrotechnik der FernUniversität Hagen Erfahrungen in der elektronischen Fernlehre gesammelt.

Seit dem Wintersemester 1995/96 wird umfangreiche Lehre über das Internet abgewickelt. Dies beinhaltet Vorlesungen, Übungen, Praktika aber auch Seminare, Prüfungen und Kommunikation mit den Dozenten und unter den Studenten im Netz.

Der zentrale Vorteil der Virtuellen Universität besteht im flexiblen, orts- und zeit-

unabhängigem, niveaueingepaßtem, bedarfsorientiertem Lernen. Simulationen, Animationen und Videokommunikation spielen dabei eine zentrale Rolle.

Die verwendeten multimedialen Elemente wurden mit den Produkten „*Authorware*“, und „*Toolbook*“ oder aber manuell in „*Java*“, „*HTML*“ oder „*PDF*“ umgesetzt. Ein aktuelles Problem besteht darin, daß für jedes Produkt ein eigenes Plug-In in den Internet-Browser integriert werden muß. Ein solches Verfahren ist wenig anwenderfreundlich.

Gewünscht wird hier vielmehr ein internationaler Standard, der die umfassende Repräsentation von multimedialen Inhalten, den Plattform übergreifenden Datenaustausch, sowie die Integration von zeitabhängigen und interaktiven Elementen ermöglicht.

Die *Moving Pictures Experts Group* (MPEG), eine Untergruppe der „*International Organization for Standardization (ISO)*“ hat im Oktober 1998 den entsprechenden Standard in der Version 1 verabschiedet: Das Multimediaformat MPEG-4.

Blickt man zurück, so sind die beiden erfolgreichen Video- und Audiokompressionsstandards *MPEG-1* und *MPEG-2* Wegbereiter für das *objektbasierte* MPEG-4. *MPEG-1* wurde 1992 definiert und ist vorwiegend für CD-ROM basierte Videoanwendungen konzipiert worden, *MPEG-2* wurde zwei Jahre später Standard für das digitale Fernsehen. Auch die neue *Digital Versatile Disk DVD* verwendet die *MPEG-2* Kodierung.

¹ FTK Forschungsinstitut für Telekommunikation, FernUniversität Hagen

² Fachgebiet Kommunikationssysteme, FernUniversität Hagen

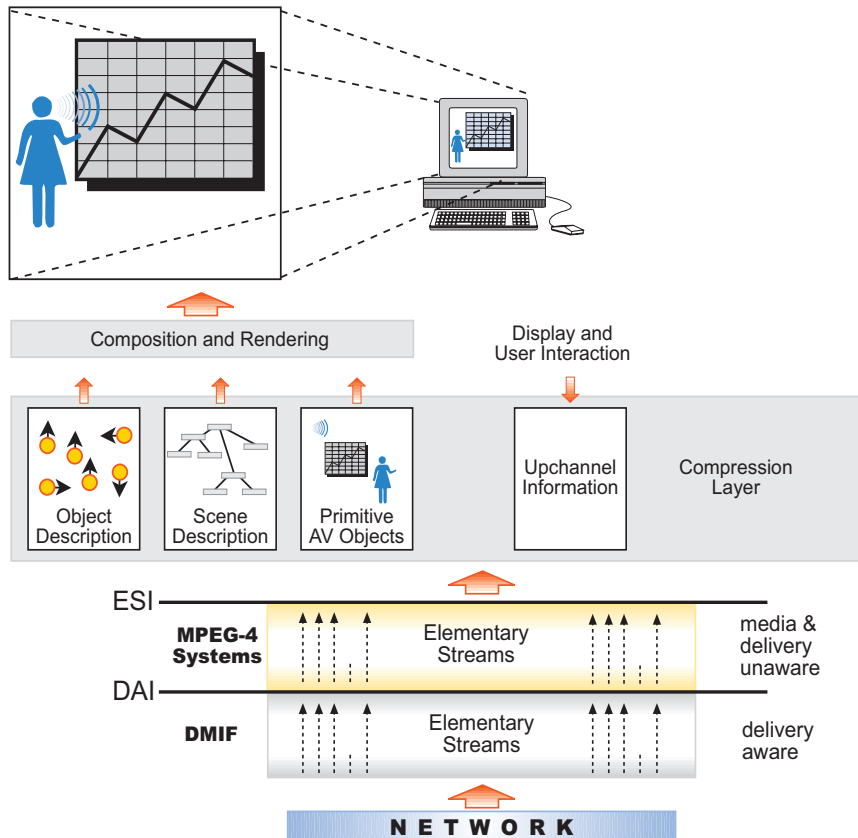


Abbildung 1: MPEG-4 Systemüberblick

Der neue Standard MPEG-4 liefert nun eine einheitliche Beschreibung für interaktive und objektbasierte Multimediaanwendungen. Neben der erhöhten Kompressionseffizienz gegenüber MPEG-1 und MPEG-2 ist der wesentliche Fortschritt bei MPEG-4 im wahlfreien Anwenderzugriff auf einzelne multimediale Elemente (*Objekte*) zu sehen. Damit ist das multimediale Objekt und dessen Arrangement, die sogenannte Szenenbeschreibung, gegenüber dem Inhalt des Objektes zentraler Gegenstand von MPEG-4.

Die einzelnen Objekte können mit anderen Kodierverfahren erstellt worden sein, so zum Beispiel Stillbilder im *JPEG* Format, Video-Konferenzmaterial mit *H.261 / H.263*, Audio mit *G.723, AC-3*, etc. Die Flexibilität von MPEG-4 erlaubt zudem die Integration von zahlreichen interaktiven Elementen, die heutzutage auch in Autorenwerkzeugen mit proprietären Lösungen zu finden sind.

Ein markantes Beispiel für die Integrität des

MPEG-4 Standards ist, daß dieser den Standard zur *Virtual Reality Modelling Language VRML* komplett beinhaltet.

Wie in **Abbildung 1** dargestellt, ist ein MPEG-4 System aufgeteilt in die Schichten *Kompression, System* und *Transport*.

Das DAI (*DMIF Application Interface*) stellt eine unabhängige Schnittstelle für den Datentransport über Netzwerke oder lokale Speichermedien dar. Das ESI (*Elementary Stream Interface*) rekonstruiert die einzelnen Multimediaobjekte. Die Kompressionsschicht wertet die Szenenbeschreibung unter Verwendung der Objektdeskriptoren aus, instanziiert den objektspezifischen Dekoder und liefert die Daten an die Kompositionsschicht (*Composition and Rendering*) aus. Die Szenenbeschreibung liefert die Position, Art und Erscheinung des multimedialen Objektes, wie Audio, Video, Stillbild. Ebenfalls werden Benutzeraktionen gemäß der Szenenbeschreibung ausgewertet. Die Kom-

position arrangiert die Objekte auf dem Benutzerbildschirm je nach Blickwinkel bzw. Vorder- und Hintergrund. Die Kompressionsschicht regelt das Zusammenspiel der Szenenbeschreibung und der Benutzerinteraktivität.

Das Prinzip, einzelne Multimedia-Elemente als Multimedia-Objekte zu verstehen und zu behandeln, ist wegen des oben beschriebenen Einsatzes von multimedial angereicherten Vorlesungen an der Virtuellen Universität hinlänglich bekannt. Heutige Autorenwerkzeuge unterstützen diese sinnvolle Sichtweise.

Das neue an MPEG-4 ist das international festgelegte Verhalten von Elementen und die Repräsentation in einer einheitlichen Beschreibungsschichtform.

2. MPEG-4 Delivery Modell

Im Teil 6 des MPEG-4 Standards ist ein neuer Übertragungsstandard für den Transport von multimedialen Objekten erarbeitet worden. Dieser behandelt ausschließlich Fragen der multimedialen Datenübertragung und wird *Delivery Multimedia Integration Framework*, kurz *DMIF* genannt.

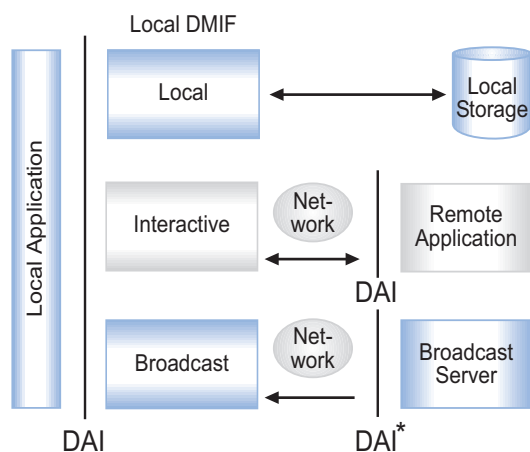


Abbildung 2: Architektur einer MPEG-4 DMIF-Kommunikationslösung für unterschiedliche Szenarien

Eine zentrale Idee bei DMIF ist, daß Applikationen immer dasselbe *DMIF Application Interface* verwenden, ohne einen Hinweis

auf die tatsächlich verwendete Quelle zu erhalten. Damit verbirgt DMIF hinter allgemeinen Merkmalen die spezielle Transporttechnologie. Alle Dienste agieren transparent und DMIF sorgt für die korrekte Adaption des speziellen Netzwerkdienstes. DMIF kann auch bei Nicht-MPEG-4 Applikationen zum Einsatz kommen.

Die **Abbildung 2** zeigt drei verschiedene DMIF Szenarien:

- *Lokale Datenablage* - bidirektional mit Kanalemulation
- *Remote-Interactive Network* - bidirektional mit transparentem Transportnetzwerk durch symmetrisches DAI
- *Broadcast* - unidirektional mit Rückkanalemulation mit noch zu spezifizierendem Server-DAI (DAI*).

Weiterhin ist DMIF so konzipiert worden, daß es mehr als einen gewöhnlichen Transportmechanismus (OSI Schicht 4) darstellt. DMIF stellt zuverlässige Point-to-Multipoint Verbindungen für Echtzeitdaten und mit der bevorzugten und skalierbaren Dienstgüte (*Quality of Service - QoS*) her. Um eine bestimmte QoS sicherzustellen, erlaubt das DAI der Anwendung, *Elementary Streams (ES)* zu Diensten zusammenzufassen und die QoS-Anforderungen für einen gewünschten Elementary Stream bzw. eine Gruppe von Elementary Streams zu spezifizieren. Diese Aufgabe wird innerhalb des DMIF Konzeptes vom FlexMux Werkzeug übernommen.

3. Das Broadcast Szenarium

In Broadcast Umgebungen wie Satellitenstrecken, Fernseekabelnetzwerken oder terrestrischen Übertragungsverbindungen existiert im allgemeinen nur ein einseitiger Informationsfluß vom Sender zum Empfänger. Dieses unidirektionale Szenarium beinhaltet die weit verbreiteten Verteilungsstrategien für Informationsdienste (*Push-Dienste*). Die **Abbildung 3** zeigt eine mögliche Broadcast Anwendungsumgebung mit einem MPEG-4

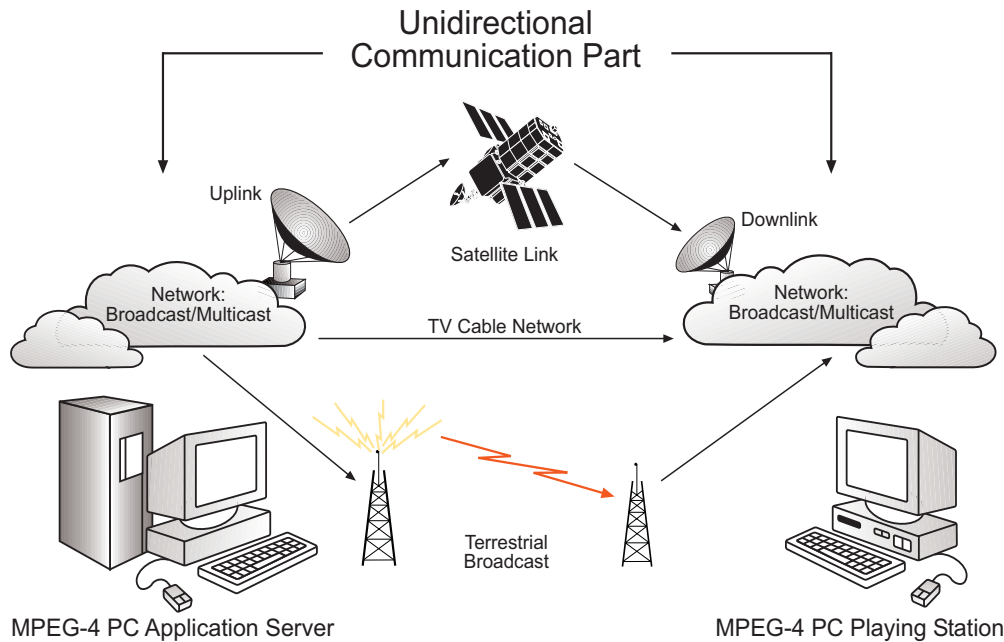


Abbildung 3: Mögliche Netzstruktur für ein MPEG-4 DMIF-Broadcast Szenarium

Server und im allgemeinen mehreren MPEG-4 Playern.

Das Broadcast Szenarium bietet sich als Grundlage für die breite und preiswerte Distribution von Lehrinhalten im Rahmen der „*Virtuellen Universität - ET-Online*“ an. Die Erweiterungsmöglichkeit auf multicastfähige Netze wurde bei der Entwicklung und Implementierung bereits mit berücksichtigt.

4. Broadcast - Realisierungskonzept

Ein Broadcast Server versendet MPEG-4 audio-visuelle Inhalte in einer oder in mehreren Sitzungen. Eine Sitzung besteht gewöhnlich aus einem Satz von Elementary Streams, die das DAI an den Übermittlungsteil DMIF weiterleitet.

Wie in **Abbildung 4** skizziert übernehmen die *DMIF Demons* die Netzwerk-Netzwerk Signalisierung (*DMIF Network Interface DNI*) mit dementsprechenden DNI Meldungen, sogenannten *DNI-Primitive*. Mit diesen vereinbaren die Demons Sitzungen und Kanäle für die Datenebene, die das entsprechende Transportprotokoll für ein spezifi-

ches Transportnetzwerk benutzen soll.

Das FlexMux Werkzeug kann korrelierte Objektströme in einen Transportkanal multiplexen. Das TransMux Werkzeug verschachtelt Datenströme optimal in Protokolleinheiten des Transportnetzwerkes.

Eine Broadcast DMIF Implementierung hat als besondere Herausforderung die unidirektionale Signalisierung, die Verwaltung von Statustabellen für den ersten Objektdeskriptor, momentan ablaufende Elementary Streams und aktivierte Transportkanäle zu bewältigen.

Seitens des MPEG-4 Players stehen die gleichen Werkzeuge FlexMux, TransMux und Demon zur Verfügung. Hier muß der DMIF Demon allerdings sowohl die eingehenden DNI Meldungen (*Indications*) als auch die lokalen Playeranforderungen (*Requests*) abhandeln können. Szenarien, die mit *Remote-Interaktive Networks* arbeiten, ist der Datenfluß vom Server zum Player selbst unidirektional, der Fluß der Signalisierungsmeldungen jedoch bidirektional. Da dieser bidirektionale sogenannte *Kontrollfluß* in unidirektionalen Broadcast Szenarien fehlt, ergeben sich nun eine Reihe von Anforderungen an die

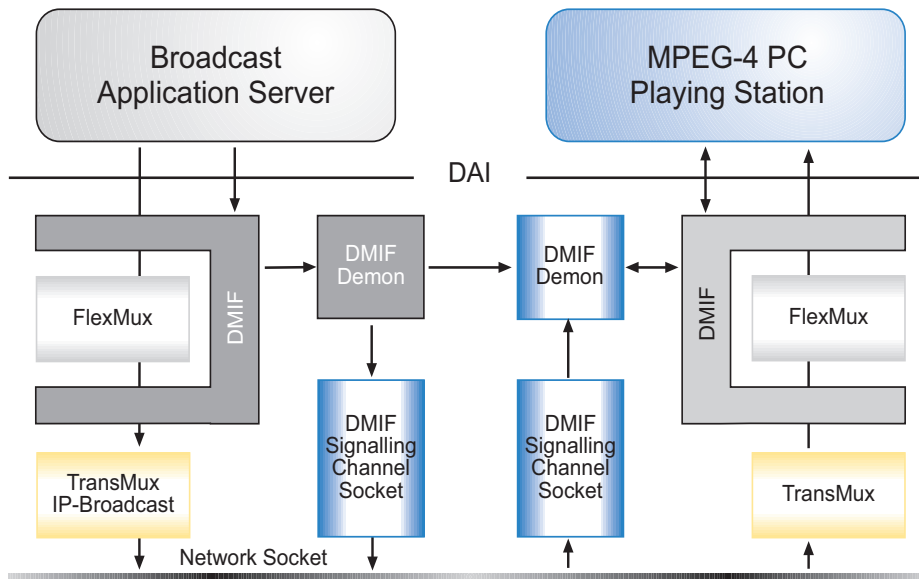


Abbildung 4: MPEG-4 DMIF Design für das Broadcast Szenarium

Implementierung. Die Lösung von zwei ausgewählten Problemstellungen, die *unidirektionale Signalisierung* und das sogenannte *Late Tuning-In*.

Unidirektionale Signalisierung

Für interaktive Netzwerke sind die *DNI-Primitive* standardisiert. Die netzwerkspezifischen Demons bilden die DNI-Primitive auf entsprechende Signalisierungen des Netzwerkes, wie *SocketConnect*, ab.

Eine Abbildung der standardisierten DNI-Primitive auf zugrundeliegende Netzwerk-Signalisierungen wurde im sogenannten *DMIF Default Signalling Protocol (DDSP)* festgelegt. Dieses Protokoll stellt ein Sitzungsverwaltungsprotokoll für den Transport von Multimediaelementen über allgemeine Übertragungstechnologien zur Verfügung. Das Protokoll umfaßt Primitive, die unter anderem Sitzungen und einzelne Datenkanäle auf- und abbauen. Im Gegensatz zu den interaktiven Szenarien wurden in DMIF für das Broadcast Szenarium bisher keine DNI-Primitive standardisiert. Im Rahmen des *MPEG-4 PC* Projektes [1] wurden für den Transport von Daten per Broadcast neue, unidirektionale Signalisierungsmeldungen von den Autoren für die im folgenden erklärte Gruppenkommunikation

entwickelt.

Ankündigung eines Dienstes

Der Broadcast Server (*Target Terminal*) kündigt eine Sitzung und den damit verbundenen Dienst wie in **Abbildung 5** skizziert an. Bei der Dienstankündigung wird ebenfalls der später beschriebene *Initial-ObjectDescriptor IOD* in dem User-User-Datenfeld übermittelt.

Der Player kann sich auf den soeben angekündigten Dienst in der entsprechenden Sitzung aufschalten. Aufgrund des IOD (MPEG-4 spezifisch), kann sich der Player (*Originator*) die aktuelle *Elementary Stream Tabelle* aufbauen.

Die Zielsetzung bei der Entwicklung der DNI Broadcast Meldungen ging dahin, einen Meldungsablauf ähnlich dem interaktiven Szenarium und die Kodierung/Dekodierung der Meldungen inklusive Parameter weiter verwenden zu können.

Kanalankündigung

Die Elementary Streams werden zum einen vom Broadcast Server angekündigt und zum anderen schaltet sich der Player auf diese entsprechenden Kanäle auf. Dieser Sachverhalt ist in **Abbildung 6** dargestellt. Die Parameter für den reinen Datentransportkanal

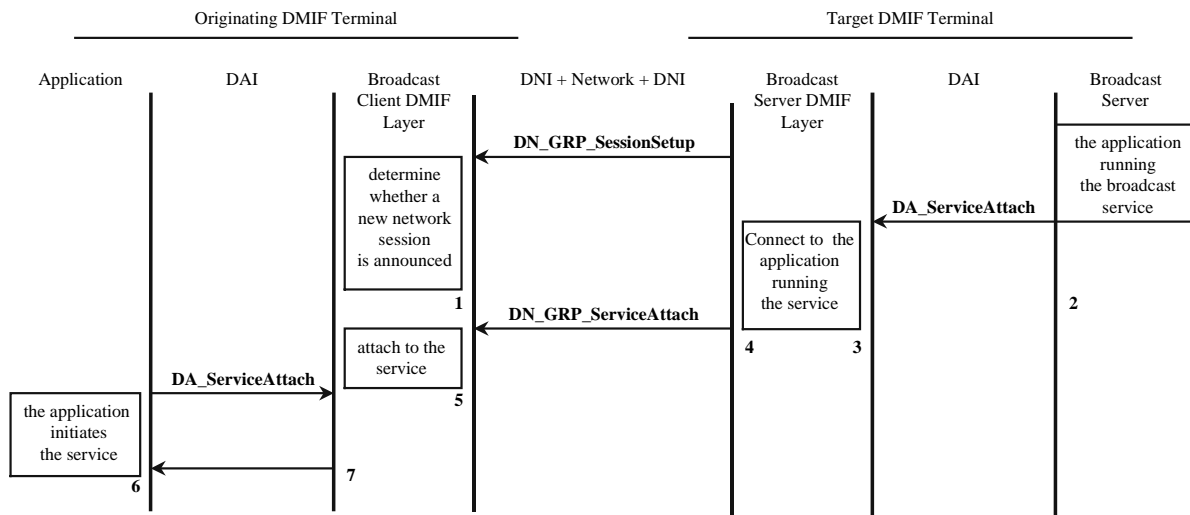


Abbildung 5: Dienstankündigung

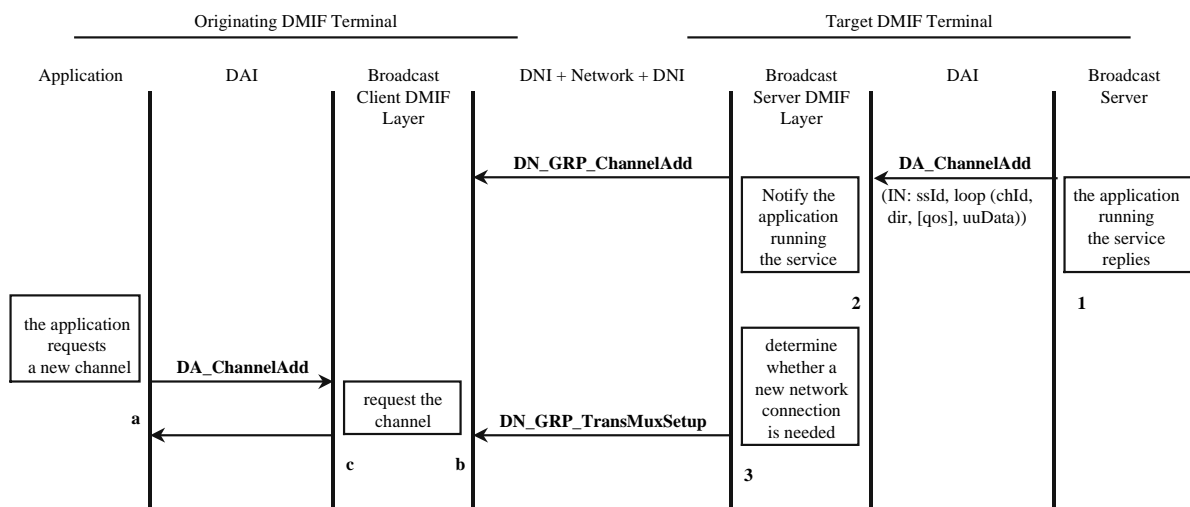


Abbildung 6: Kanalankündigung

werden in der *TransMuxSetup* Nachricht übermittelt. Damit kann auf beiden Seiten die eigentliche Socket-Verbindung aufgebaut werden.

Analog zum Sitzungsaufbau und Kanalaufbau geschieht der jeweilige Abbau.

Late Tuning-In

Zur Konstruktion der Szene muß die Szenenbeschreibung (*Binary Format for Scene Description - BIFS*) an den Player übermittelt werden. Der IOD ist zum Auffinden der Objekte und der Szene unerlässlich. Damit sich die Player in einem Broadcast Szenarium während einer laufenden Übertragung zu beliebigen Zeitpunkten aufschalten können, müssen alle einmalig benötigten Informationen, wie der IOD, Hintergrundbilder usw.

periodisch wiederholt werden. Dafür wird im Server eine sogenannte *Karusselltechnik* eingesetzt, die das gewünschte *Late Tuning-In* des Players ermöglicht.

5. Zusammenfassung

Die Ergebnisse der Untersuchungen zeigen, daß die Nutzung von MPEG-4 konformen Anwendungen innerhalb einer reinen Broadcast Infrastruktur prinzipiell möglich ist. Erkauft wird dieses Szenarium allerdings durch eine Karusselltechnik, die eine Erhöhung der Datenrate auf dem Übertragungskanal mit sich bringt. Es ist weiterhin zu berücksichtigen, daß das Aufschalten auf einen Dienst respektive Kanal im Zusammenhang

mit den entsprechenden Ankündigungen unter Umständen Timing Probleme verursacht und damit besondere Anforderungen an die Performance des Player PCs stellt. Eine günstigere Lösung wäre die konsequente Umsetzung der Push-Technologie bis in den Player selbst hinein. Dieses Verfahren erfordert strukturelle Änderungen der Playerimplementierung, die derzeit auf das Remote-Interactive Szenarium optimiert ist. Es wird derzeit diskutiert, inwiefern entsprechende Verbesserungen eventuell Ziel zukünftiger Erweiterungen des Standards werden könnten.

Der besondere Dank der Autoren gilt Herrn Prof. Dr.-Ing. Firoz Kaderali³ für die Anregung und Unterstützung der Arbeiten sowie Frau Farkhondeh Mehdizadeh und Herrn Mostafa Dashti für die Entwicklung von Softwarekomponenten im Rahmen ihrer Diplomarbeiten.

Literatur

- [1] EU Esprit-Projekt 23191 “MPEG-4 PC – MPEG-4 System Implementation and Tools for PC”, Nov. 96 – Aug. 99
- [2] Die Virtuelle Universität – ET-Online an der FernUniversität GH in Hagen. Webpage: <http://virtuelle-uni.fernuni-hagen.de>
- [3] ISO/IEC JTC/SC29/WG11: “Information Technology – Generic Coding of Audio - Visual Objects – Part 1: Systems”, ISO/IEC IS 14496-1, 1999
- [4] ISO/IEC JTC/SC29/WG11: “Information technology – Generic Coding of Audio - Visual Objects – Part 2: Visual”, ISO/IEC IS 14496-2, 1999
- [5] ISO/IEC JTC/SC29/WG11: “Information technology – Generic Coding of Audio - Visual Objects – Part 3: Audio”, ISO/IEC IS 14496-3, 1999
- [6] ISO/IEC JTC/SC29/WG11: “Information technology – Generic Coding of Audio - Visual Objects – Part 6: Delivery Multimedia Integration Framework”, ISO/IEC IS 14496-6, 1999
- [7] ISO/IEC: “Information technology - Open Systems Interconnection - Basic Reference Model: The Basic Model”, ISO/IEC 7498-1, 1994
- [8] ISO/IEC JTC/SC29/WG11: M4182: Stepping, M., et al.: “DMIF Application Interface: Syntax Definition”, 1998
- [9] ISO/IEC JTC/SC29/WG11: M4191: Stepping, M.: “DAI Syntax definition: Source files, C++, Java“, 1998
- [10] ISO/IEC JTC/SC29/WG11: M4507: Stepping, M.: “DAI Syntax definition-several additions V1/V2“, 1999
- [11] ISO/IEC JTC/SC29/WG11: M4508: Stepping, M.: “DMIF Group and Broadcast Signalling: comments and DNI messages“, 1999
- [12] ISO/IEC JTC/SC29/WG11: M4507: Stepping, M.: “Monitoring DMIF“, 1999

³ Fachgebiet Kommunikationssysteme,
FernUniversität Hagen