

Methodik und Software zur Erstellung und Konsum von MPEG-21 Digital Items

Harald Kosch and Alexander Arrich

Institut für Informationstechnologie, Universität Klagenfurt, Österreich
harald.kosch@itec.uni-klu.ac.at
<http://www-itec.uni-klu.ac.at/~harald/MPEG/>

Abstract Im MPEG-21 Multimedia Framework Standard spielt das Digital Item als fundamentale Transaktions- und Austauschseinheit eine zentrale Rolle. Dieser Artikel beschreibt eine Methodik und Software zur Erstellung und den Konsum von Digital Items. Das Softwarewerkzeug setzt sich aus zwei Teilen zusammen: Der *DI Builder* erlaubt es Benutzern MPEG-21 Digital Items zu erstellen, die mit dem *DI Consumer* konsumiert werden können. Der Konsum eines Digital Items umfasst das Abspielen von Mediendateien und die Betrachtung der Metadaten eines Digital Items. Die Software demonstriert Teile des MPEG-21 Standards, im speziellen Teil 2-Digital Item Declaration, Teil 3-Digital Item Identification, Teil 5-Rights Expression Language, Teil 6-Rights Data Dictionary und Teil 7-Digital Item Adaptation.

Schlagerworte: MPEG-21, Verteilte Multimedia-Systeme, MPEG-7.

1 Einleitung

MPEG-21 [1] ist der fünfte Standard der Moving Pictures Experts Group (MPEG). Während es sich bei MPEG-1 [2], MPEG-2 [3] und MPEG-4 [4] um reine Kodierungsstandards handelt, hat MPEG mit MPEG-7 [5] einen Beschreibungsstandard für Multimedia Inhalte entwickelt¹. Mit MPEG-21 will man nun ein so genanntes 'Multimedia-Framework' schaffen, welches das Austauschen von Multimedia-Daten und Inhalten interoperabel, benutzerfreundlich und fairer macht [6].

Digital Items (DIs) spielen im MPEG-21 Multimedia Framework Standard als Transaktions- und Austauschseinheit eine zentrale Rolle. In diesem Artikel werden eine Methodik und Software für die Erstellung und den Konsum von MPEG-21 Digital Items beschrieben. Die Erstellung von Digital Items basiert auf der partiellen Abbildung der Digital Item Declaration Language (DIDL) auf eine Objektstruktur, welche die Erstellung, Verarbeitung und Benutzung von Digital Items erstmalig ermöglicht. Die zwei entwickelten Softwarewerkzeuge *DI Builder* und *DI Consumer* erlauben die Erstellung bzw. den Konsum von Digital Items, wobei der Konsum das Abspielen von Mediendateien und das Betrachten der Metadaten des Digital Items umfasst.

MPEG-21 ermöglicht in einem verteilten Multimediasystem den Datenaustausch über Plattformgrenzen und Anwendergruppen hinweg. Eine Reihe von Beschreibungs-, Kodierungs-, Austausch- und Sicherheitsstandards von multimedialen Medien und Inhalten sind realisiert oder in der Definitionsphase und garantieren die Interoperabilität und Funktionalität. MPEG-21 ist als ein offenes System entwickelt worden, welches bestehende Standards integriert (u.a. unterschiedliche Kodierungsstandard und Metadatenstandards wie Dublin Core [7]), Freiraum für unterschiedliche technische Realisierungen lässt und die Heterogenität in den Fähigkeiten von Netzwerken und Terminals überbrückt.

MPEG-21 ist ein relativ junger MPEG Standard, viele Teile werden oder wurden in diesem Jahr veröffentlicht (u.a. Definition eines Digital Items, Ressourcenadaptierung, Rechtesprache), andere befinden sich in Arbeit (u.a. Intellectual Property Management)². Für einige Teile von MPEG-21 gibt es kleine Referenzimplementierungen, aber es gab bisher keine Methodik Software, mit der sich Digital Items erstellen, verarbeiten und konsumieren lassen. Eine manuelle Erstellung der XML-Daten mit Hilfe gängiger Editoren erweist sich natürlich als mühsam und vor allem fehleranfällig, insbesondere dann, wenn extern vorhandene Ressourcenbeschreibungen eingefügt werden müssen.

¹ Überblick über die MPEG-Standards unter: <http://www.chiariglione.org/mpeg/standards.htm>

² siehe auch <http://www.chiariglione.org/mpeg/workplan.htm>

In diesem Zusammenhang stellen wir unsere Softwarewerkzeuge **DI Builder** und **DI Consumer** vor, welche es ermöglichen, Digital Items schnell und effektiv zu erstellen und zu konsumieren. Zur Erstellung von Digital Items im DI Builder wurde die *Digital Item Declaration Language (DIDL)* auf eine Objektstruktur abgebildet. Jedes Objekt dieser Struktur entspricht einem item-Element des Digitalen Items und bietet Methoden zur Erstellung und Abfrage eines korrespondierenden DIDL XML-Fragments an. Aus der Objektstruktur kann direkt ein Digital Item in XML Notation abgeleitet werden. Die Verarbeitung eines Digital Items im DI Consumer geschieht durch so genannte Engines. Die Aufgabe einer Engine ist die Übernahme und Verarbeitung eines Digital Items mit oder ohne Hilfe des Benutzers. Die Verarbeitung der Digital Items umfasst neben dem Abspielen der in ihm enthaltenen Mediendateien auch das Betrachten der assoziierten Metadaten in einem XML Viewer.

In diesem Sinne stellt unser System eine Weiterentwicklung existierender komponenten-basierter Systeme dar ; einige wichtige Systeme werden in unserer Zusammenstellung [8] behandelt. Zum einen wird das erste Mal auf MPEG-21 zum Austausch von Medien und Inhalte zurückgegriffen, zum anderen verknüpft das System zum ersten Mal wichtige Aspekte wie Rechte, Retrieval, Adaptierung und Darstellung in einem Framework. Unsere Architektur ist so allgemein gehalten, dass weitere Engines ohne Veränderung des Gesamtframeworks integriert werden können.

Weiterer Aufbau des Artikels: Im nächsten Teil wird der MPEG-21 Standard beschrieben, wobei schwerpunktmäßig auf die in der Arbeit verwirklichten Teile eingegangen wird. Danach erfolgt die Beschreibung des Peer-to-Peer-Anwendungsszenarios, das durch die beiden Softwareprodukte unterstützt wird. Im vierten Teil wird die Methodik und Software zur Erstellung von Digital Items beschrieben, gefolgt von der Beschreibung für den Konsum von Digital Items. Den Abschluss bilden die Schlussfolgerungen aus der Arbeit und Anregungen für weitere Entwicklungen.

2 MPEG-21 Standard

Die Lieferkette für Multimediainhalte umfasst die Erzeugung, Produktion, Lieferung und den Konsum von Multimediainhalten [6]. Für jeden einzelnen Schritt der Lieferkette existieren Lösungen bzw. sind Lösungen gerade in Entwicklung. Jedoch existierte bisher kein Gesamtbild, das die einzelnen und separaten Schritte zusammenfügt hätte. **MPEG-21** entwickelt ein Multimedia-Framework zur Integration dieser zahlreichen Elemente der Multimedialieferkette, welches von allen Anwendergruppen auf die gleiche Weise nutzbar ist. Zur Realisierung dieses einheitlichen Systems definierte MPEG folgende Kernelemente: Das **Digital Item** und den **Benutzer**.

2.1 Zentrale Konzepte von MPEG-21

Die beiden zentralen Konzepte eines MPEG-21 verteilten Multimediasystems sind das *Digital Item* und der *Benutzer* [1,6]. Das Digital Item ist die vom Benutzer verwendete Austausch- und Transaktionseinheit des Systems. Benutzer sind alle Entitäten, die im MPEG-21 System interagieren bzw. Digital Items erzeugen und verwenden.

Das *Digital Item* ist ein strukturiertes und hierarchisch aufgebautes Objekt, das Multimediaressourcen und Metadaten kapselt. Die Metadaten eines Digital Items beschreiben einerseits das Digital Item und dessen Gebrauch, andererseits aber auch die Ressourcen des Digital Items.

Der *Benutzer* ist jede Entität, die am MPEG-21 System teilnimmt bzw. Digital Items verwendet. Es fallen daher Einzelpersonen, Personengruppen, Konsumenten und Organisationen gleichermaßen unter den Begriff *Benutzer*. In diesem Sinn folgt MPEG-21 dem Peer-to-Peer-Paradigma mit gleichberechtigten Teilnehmern [9] und nicht dem Client/Server-Paradigma, bei dem Benutzer klar definierte und festgelegte Rollen einnehmen [10]. Die Rolle eines Benutzers wird durch die Interaktion mit anderen Benutzern bestimmt, das heißt beispielsweise, dass ein Benutzer in einer Transaktion die Rolle des Konsumenten und in einer anderen Transaktion die Rolle des Produzenten übernehmen kann. Abbildung 1 veranschaulicht die Beziehungen zwischen Benutzern, Digital Items und den Kernelementen des MPEG-21 Multimediasystems.

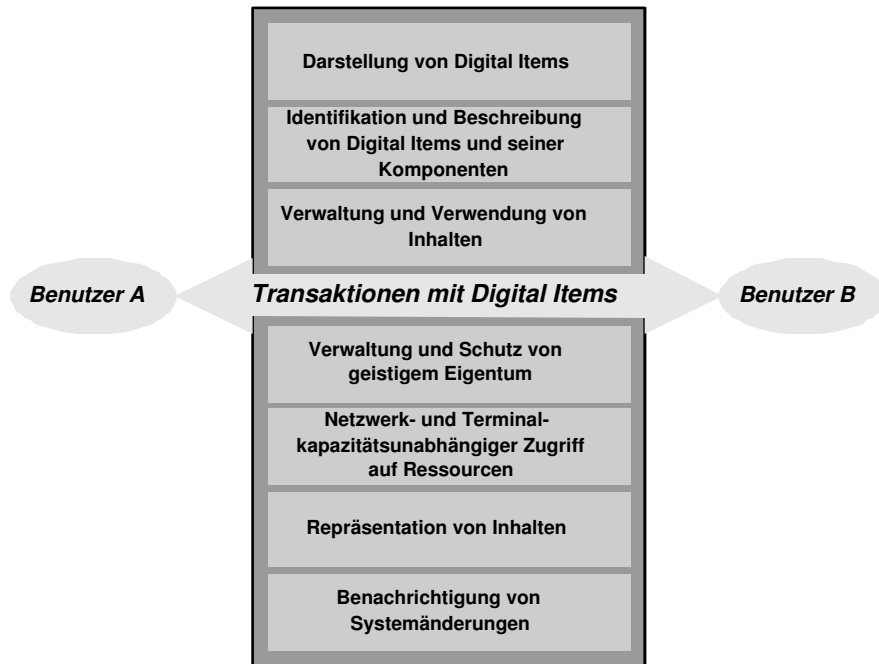


Figure 1. Die konzeptuellen Hauptelemente des MPEG-21 Standards

2.2 Teile des MPEG-21 Standards

Der MPEG-21 Standard umfasst inzwischen schon **16 Teile**, die nachfolgend zusammen mit einer kurzen Beschreibung aufgelistet sind. Die in der Arbeit verwendeten Teile werden gesondert näher beschrieben³.

- Teil 1: Visionen, Technologien und Strategien von MPEG-21.
- Teil 2: Digital Item Declaration (DID) zur Darstellung von Digital Items.
- Teil 3: Digital Item Identification (DII) zur Identifikation von Digital Items und seiner Komponenten.
- Teil 4: Intellectual Property Management and Protection (IPMP) zum Schutz von Inhalten und den damit verbundenen Rechten (in Bearbeitung).
- Teil 5: Rights Expression Language (REL) zur Spezifikation von Rechten in Bezug auf die Verwendung von Multimediainhalten.
- Teil 6: Rights Data Dictionary (RDD) stellt eine Menge an Begriffen bereit, die von der REL verwendet werden.
- Teil 7: Digital Item Adaptation (DIA) zur Anpassung von Ressourcen und Metadaten an verfügbare Netzwerk- und Terminalkapazitäten.
- Teil 8: Referenzsoftware zur Validierung der Vorschläge für die Teile 1-7.
- Teil 9: Ein Dateiformat zum effizienten Speichern und Zugriff auf Digital Items (in Bearbeitung).
- Teil 10: Digital Item Processing umfasst eine Menge von Operationen zur Konfiguration und Manipulation von Digital Items (in Bearbeitung).
- Teil 11: Persistent Association Tools umfasst Methoden und Techniken zur dauerhaften Verknüpfung von Informationen und ihrer Identifikation und zur Beschreibung von Inhalten mit den Inhalten (z.B. Wasserzeichen) selbst (in Bearbeitung).
- Teil 12: Resource Delivery Test Bed ist eine Testumgebung für MPEG-21 Anwendungen.
- Teil 13: Scalable Video Coding sucht Techniken und Methoden zur skalierbaren Kodierung von Videoströmen (in Bearbeitung).
- Teil 14: Conformance Testing (in Bearbeitung).
- Teil 15: Event Reporting für die Meldung von Systemänderungen (in Bearbeitung).
- Teil 16: Binary Format zur kodierten Darstellung von MPEG-21 Digital Items (in Bearbeitung).

³ siehe auch http://www.chiariglione.org/mpeg/working_documents.htm

```

<?xml version="1.0" encoding="UTF-8"?>
<DIDL xmlns:targetNamespace="urn:mpeg:mpeg21:2002/01-DIDL-NS"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <ITEM>
    <DESCRIPTOR>
      <STATEMENT>Spiderman-Film in unterschiedlichen Auflösungen</STATEMENT>
    </DESCRIPTOR>
    <ITEM>
      <DESCRIPTOR>
        <STATEMENT>Spiderman-Film in niedriger Auflösung für einen schnellen
          Download
        </STATEMENT>
      </DESCRIPTOR>
      <COMPONENT>
        <DESCRIPTOR>
          <STATEMENT>Spiderman-Film in Auflösung 320x240</STATEMENT>
        </DESCRIPTOR>
        <RESOURCE REF="www.online-films/spiderman_low.mpg"/>
      </COMPONENT>
    </ITEM>
    <ITEM>
      <DESCRIPTOR>
        <STATEMENT>Spiderman-Film in sehr hoher Auflösung</STATEMENT>
      </DESCRIPTOR>
      <COMPONENT>
        <DESCRIPTOR>
          <STATEMENT>Spiderman-Film in DVD - Qualität</STATEMENT>
        </DESCRIPTOR>
        <RESOURCE REF="www.online-films/spiderman_high.mpg"/>
      </COMPONENT>
    </ITEM>
  </ITEM>
</DIDL>

```

Figure 2. Einfaches Digital Item

Digital Item Declaration (DID) Die Digital Item Declaration Language (DIDL) ist das wichtigste Ergebnis der DID Arbeitsgruppe und stellt ein XML Schema dar, das die Struktur von Digital Items bestimmt. Die Bestandteile eines Digital Items sind Ressourcen und Metadaten. Die DIDL bestimmt, wie diese Ressourcen und Metadaten auf eine strukturierte Weise zu einem XML-basierten Digital Item integriert werden. Abbildung 2 zeigt ein einfaches Beispiel. Das DIDL-Element ist das Wurzelement eines Digital Items. Es beinhaltet in einfacher Form ein oder mehrere *item-Elemente*, welche die Grundeinheit einer zusammenhängenden MPEG-21 Beschreibung darstellen. Diese können hierarchisch angeordnet werden. Ein item-Element enthält neben Beschreibungskomponenten (*Descriptor-Elemente*, z.B. MPEG-7 oder Dublin Core Beschreibungen) ein *Component-Element*, welches die Ressource als Referenz spezifiziert oder inline trägt. Digital Items lassen sich durch *Auswahlstrukturen* konfigurieren. Kapitel 4.1 zeigt ein einfaches Beispiel.

Digital Item Identification (DII) Die Digital Item Identification (DII) Spezifikation erlaubt die Assoziation von eindeutigen Bezeichnern mit Digital Items bzw. Teilen eines Digital Items unabhängig vom Typ und der Granularität des spezifischen Teils. Zur Identifikation von Digital Items bzw. Komponenten eines Digital Items nützt DII die Uniform Resource Identifiers. Zur Differenzierung zwischen XML Schemata nützt DII den XML Namespace-Mechanismus. Unterschiedliche Arten von Digital Items wie Content Digital Items und Method Digital Items werden unterschieden, indem der Typ des Digital Items angegeben wird. Ein weiteres Entwurfsmerkmal von MPEG-21 DII ist, dass nicht versucht wurde, bestehende Identifikationssysteme zu ersetzen, sondern dass deren Nutzung explizit ermöglicht wurde.

Rights Expression Language (REL) Die Rights Expression Language (REL) ist eine Sprache zur Darstellung von Rechten an digitalen Inhalten und an Metadaten. Die REL benutzt zur Beschreibung von Rechten Begriffe aus dem Rights Data Dictionary (MPEG-21 RDD), einer erweiterbaren Sammlung von Definitionen [11]. Durch die Rights Expression Language können digitale Inhalte geschützt und Rechte,

Bedingungen und Gebühren für die Verwendung digitaler Inhalte spezifiziert werden. Das Kernelement der REL ist die Berechtigung (Grant), die es einem Rechteinhaber bzw. -herausgeber erlaubt, einem Rechteinhaber Rechte in Form von erlaubten Aktivitäten an einer Ressource zuzuerkennen. Die Ausübung von Rechten durch den Rechteinhaber kann von Bedingungen, die der Rechteinhaber definiert, abhängen. Das REL Datenmodell wurde als eigenes XML Schema verwirklicht.

Digital Item Adaptation (DIA) Ein Ziel des MPEG-21 Multimediasystems ist der transparente Zugriff auf Multimediainhalte, unabhängig von Netzwerk- und Terminalkapazitäten. Dieses Ziel erfordert, dass Digital Items bzw. die darin gekapselten Ressourcen und Metadaten beim Erkennen von Kapazitätsengpässen angepasst werden können. Bei der Adaptierung von Digital Items werden die Ressourcen und Metadaten durch Digital Item Adaptation Tools an gegebene Netzwerk- und Terminalkapazitäten angepasst. Dabei wird ausgehend vom Originalitem ein adaptiertes Ergebnisitem erzeugt, indem die Ressourcen in einer Resource Adaptation Engine und die Metadaten in einer Description Adaptation Engine angepasst werden. Der Anpassungsprozess wird in beiden Engines durch DIA Tools gesteuert [12]. Die DIA Tools umfassen Beschreibungen und formatunabhängige Mechanismen zur Steuerung des Anpassungsprozesses.

Ein Beispiel eines DIA Tools ist die von uns entwickelte standardisierte Generic Bitstream Definition Language (gBSDL) [13]. Die gBSDL beschreibt die Struktur einer Ressource (z.B. die Frames eines Videos) in einer generischen Weise, wobei diese Beschreibung in XML ausgedrückt wird. Eine Adaptation Engine verwendet diese Beschreibung als Grundlage für die Anpassung eines Videos, beispielsweise zum Entfernen von Frames, um die Bandbreitenanforderung zu reduzieren. Der DI Consumer beinhaltet eine Adaptation Engine.

Beispiele für weitere DIA Tools sind Beschreibungen des Benutzers, der Gebrauchsumgebung, des Netzwerkes und der Fähigkeiten von Endgeräten. Diese Beschreibungen werden dazu verwendet, um den Adaptierungsprozess zu steuern. In unserem Framework kann der Benutzer den Adaptierungsprozess mit Hilfe einer Auswahlliste möglicher Adaptierungsmethoden direkt steuern.

2.3 MPEG-7 Beschreibung im MPEG-21 Digital Item

MPEG-7 ist ein MPEG Standard zur Beschreibung von audiovisuellen Inhalten durch Metadaten [5]. Es können sowohl syntaktische und semantische Eigenschaften von Multimediadaten durch Metadaten beschrieben werden. Die beschreibbaren Eigenschaften von audiovisuellen Inhalten sind im Standard in Form von XML Schema-Konstrukten, so genannten Deskriptoren, definiert. Ein Beispiel einer syntaktischen Eigenschaft ist etwa die Beschreibungen der Farbverteilung eines Bildes, ein Beispiel einer semantischen Eigenschaft ist die Beschreibung der Objekte und ihrer Beziehungen, die im Bild sichtbar sind. Zu bemerken sei noch, dass der Standard zwar die XML Schema-Konstrukte normativ vorgibt, die Extraktion und der Ähnlichkeitsvergleich von Beschreibungen aber als Gegenstand des Firmenwettbewerbs belassen wurden.

Die übliche Methode, externe Metadaten in Digital Items einzubinden ist, die Metadaten als DIDL Deskriptor-Element ins Digital Item einzufügen. Wie der Name impliziert, ist das DIDL Deskriptor-Element die primäre Wahl, Digital Items bzw. deren Ressourcen zu beschreiben. Beispiele für Beschreibungen sind etwa Ressourcenbeschreibungen, Lizenzen, Bezeichner und ähnliches. In unserer Arbeit haben wir beispielsweise den MPEG-7 GoF/GoP-Deskriptor als Ressourcenbeschreibung implementiert. Der MPEG-7 Group-of-Frames/Group-of-Pictures (GoF/GoP)-Deskriptor erstellt ein HSV-Farbhistogramm für eine Gruppe von (Video-)Frames, beispielsweise die Frames einer abgeschlossenen Szene [14]. Er wurde beispielhaft verwendet, um ein Content-Based Retrieval (CBR) auf Digital Items zu ermöglichen und Ähnlichkeitsabfragen zwischen Videos und zwischen Videos und Bildern zu unterstützen [15]. Unsere Softwarelösung stellt die notwendigen Engine-Container für CBR auf den GoF/GoP-Deskriptoren zur Verfügung, auf die dann entsprechende Retrieval- und Datenbankanwendungen, wie z.B. unsere MPEG-7 Multimediadatenbank [16], zugreifen können.

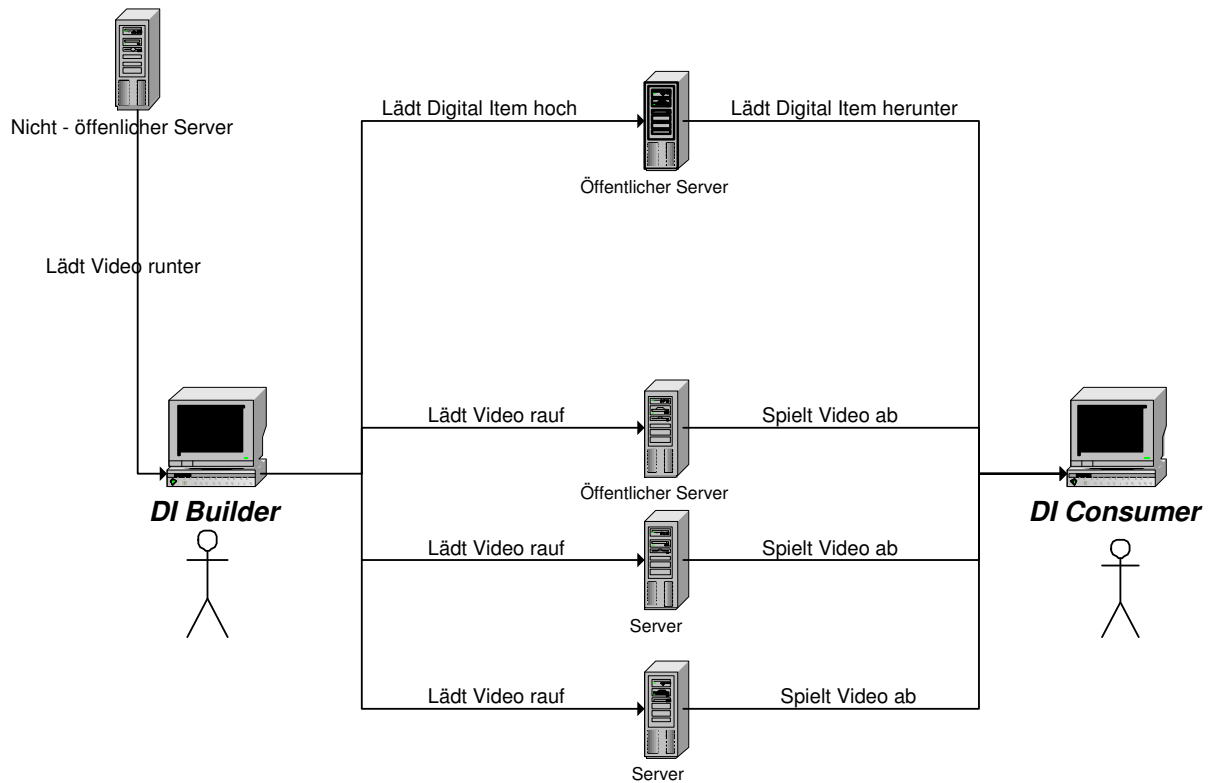


Figure 3. Anwendungsszenario mit Produzent und Konsument

3 Verteiltes Anwendungsszenario

Dieses Kapitel zeichnet ein verteiltes Anwendungsszenario der Benutzung des DI Builders und DI Konsument. Es folgt dem Peer-to-Peer-Charakter von MPEG-21, das heißt insbesondere auch, dass die Rolle eines Benutzers von der jeweiligen Interaktion mit anderen Benutzern abhängt und nicht statisch festgelegt ist. In dem hier beschriebenen Anwendungsszenario kann ein Benutzer die Rollen "Konsument" und "Erzeuger" von Digital Items annehmen.

Das Anwendungsszenario (siehe Abbildung 3) umfasst zwei Rollen, den *DI Builder* und den *DI Consumer*. Der DI Builder erstellt MPEG-21 Digital Items, die auf Videodateien⁴ ausgelegt sind. Bei der Erstellung eines Digital Items muss der DI Builder für jede Videodatei des Digital Items Nutzungsrechte vergeben. Nach der Erstellung des Digital Items lädt der DI Builder das Digital Item und dessen Videodateien auf beliebige öffentliche Server hoch, damit der *DI Consumer* auf das Digital Item und dessen Videodateien zugreifen kann. Der *DI Consumer* verarbeitet Digital Items. Nach dem Laden eines Digital Items von einem öffentlichen Server oder von einem lokalen Verzeichniss kann der Benutzer aus den Digital Items Videodateien auswählen. Wenn der Benutzer zum Abspielen einer ausgewählten Videodatei berechtigt ist, dann darf er/sie das Video abspielen und die Metadaten des Digital Items ansehen. Integriert in den *DI Consumer* ist auch noch eine *Adaptation-Engine*. Mit ihrer Hilfe kann die Ressource an Betriebsmittelengpässe angepasst werden. Dies geschieht aufbauend auf die in das Digital Item integrierten Digital Item Adaptation (DIA)-Beschreibungen.

⁴ Im DI Builder können auch andere Medientypen wie Bilder und Audio angegeben werden, solange JMF diese verarbeiten kann. Allerdings können nur für Videodateien die MPEG-7 und gBSDL-Beschreibungen erzeugt werden.

4 Methodik und Software zur Erstellung von MPEG-21 Digital Items

Der DI Builder erlaubt die einfache und transparente Erstellung von MPEG-21 Digital Items in einer graphischen Oberfläche. Die Methodik zur Erstellung von Digital Items basiert auf der partiellen Abbildung des Digital Item Declaration (DID) Items auf eine Objektstruktur und die Generierung eines Digital Items aus der Objektstruktur.

4.1 Methodik zur Erstellung von MPEG-21 Digital Items

Die Methodik für die Erstellung von Digital Items basiert auf der Abbildung des DID Items auf eine Objektstruktur, die semantisch und strukturell der Digital Item Declaration Language (DIDL) entspricht.

Die Objektstruktur (siehe Abbildung 4) umfasst Objekte für den Bezeichner des Digital Items, für Deskriptoren des Digital Items, Lizenzen, Auswahlstrukturen (DIDL Choices) und Komponenten des Digital Items, wobei Komponenten aus Videodateien sowie MPEG-7 und MPEG-21 DIA Beschreibungen der Videodateien bestehen. Die Objekte sind in der Objektstruktur den DID Items entsprechend angeordnet, wodurch aus der Objektstruktur direkt ein MPEG-21 Digital Item abgeleitet bzw. erzeugt werden kann. Jedes Objekt der Objektstruktur erstellt ein voll- und selbständiges DIDL XML-Fragment aus seinen Daten.

Das Digital Item-Objekt Das Digital Item-Objekt dient als Behälter für andere Objekte und ist das Basisobjekt der Objektstruktur. Das Digital Item-Objekt setzt die DIDL XML-Fragmente seiner Komponenten zu einem Digital Item zusammen. Das Digital Item-Objekt ist auch für die Erzeugung des Headers verantwortlich, wofür es die XML Namespaces, die von seinen Komponenten benötigt werden, abfragt. Jedes Objekt muss die XML Namespaces, die es benötigt, bekannt geben.

Das Digital Item Bezeichner-Objekt Das Digital Item Bezeichner-Objekt wird zur Erzeugung des Bezeichners eines Digital Items verwendet. Der generierte Bezeichner ist ein Uniform Resource Name (URN), dessen Zweck die dauerhafte und ortsunabhängige Bezeichnung von Digital Items ist. Der Aufbau des Bezeichners folgt folgendem einfachem Schema:

```
Bezeichner ::= "urn:mpeg21:" <String>  
String     ::= <Datum>":"<IP>":"<Zufallszahl>  
Datum      ::= <Jahr><Monat><Tag>
```

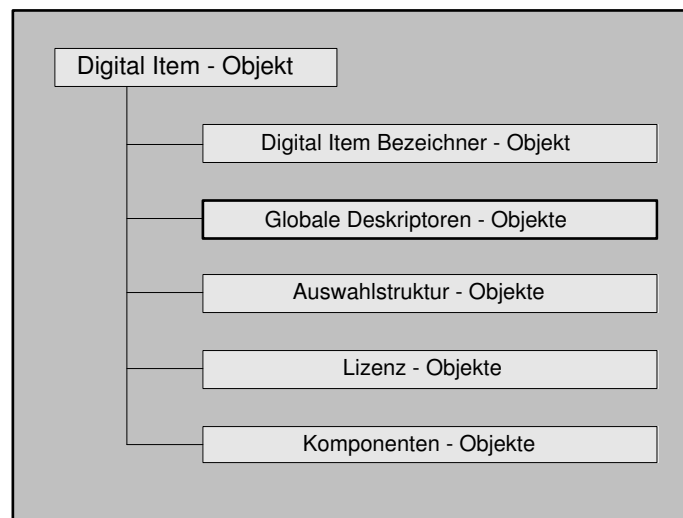


Figure 4. DID Modellkonforme Objektstruktur

Für das Datum wird die aktuelle Systemzeit des Benutzerrechners als Basis verwendet, wobei das Jahr als vierstellige Zahl, der Monat und Tag jeweils als zweistellige Zahlen dargestellt werden. Auf das Datum folgen die Stringrepräsentationen der numerischen IP-Adresse des Benutzerrechners und eine dreistellige Pseudozufallszahl. Dieses Schema zur Erzeugung von Bezeichnern kann im DI Builder durch eigene Erzeugungsschemata ersetzt werden. Das Digital Item Bezeichner-Objekt erstellt aus diesen Informationen ein DIDL Descriptor-Fragment.

Deskriptoren zur Beschreibung des Digital Items Globale Deskriptoren werden zur allgemeinen Beschreibung des Digital Items benutzt, das heißt, mit globalen Deskriptoren wird der Verwendungszweck des Digital Items in textueller Form angegeben. Ein Objekt für einen globalen Deskriptor besteht aus dem Namen, dem Typ und dem Inhalt eines globalen Deskriptors. Der Typ des globalen Deskriptors beschreibt die Ausprägungsform des Inhalts, beispielsweise reiner Text oder Information, in XML Notation. Das Objekt eines globalen Deskriptors erstellt aus dem Namen, dem Typ und dem Inhalt ein DIDL Descriptor-Fragment.

Lizenzen des Digital Items Zu jeder Videodatei eines Digital Items muss im DI Builder eine Lizenz erstellt werden. Eine Lizenz besteht aus der URL einer Videodatei, dem Rechtevergeber, dem Rechteinhaber, dem Recht und Bedingungen, denen die Ausübung des Rechts unterliegen. Als Bedingungen können der Zeitraum, in dem die Ausübung des Rechts erlaubt ist und die maximale Anzahl erlaubter Rechteaustübungen angegeben werden. Ein Lizenzobjekt erzeugt aus seinen Daten ein REL License-Fragment. Die REL License-Fragmente der Lizenzen werden zu einem REL LicenseGroup-Fragment zusammengefasst und in ein DIDL Descriptor-Fragment eingebunden.

Beispiel: Betrachten wir folgendes vom DI Builder erzeugtes REL XML-Fragment. Im ersten Teil wird der Rechteinhaber (**Edith**) mit dem keyHolder-Element⁵ spezifiziert. Danach erfolgt die Bekanntgabe ihrer Rechte, hier das Recht, die Ressource, identifiziert durch die DII `urn:harald-1`, abzuspielen. Es folgt eine Nebenbedingung, die aussagt, dass dieses Recht nur im Jahr 2004 ausgeübt werden kann. Danach identifiziert sich der Rechteinhaber (**Harald**) z.B. über seine Digitale Signatur.

```
<license xmlns="http://www.xrml.org/schema/2002/05/xrml2core"
xmlns:mx="urn:mpeg:mpeg21:2002:01-REL-NS">
  <!--Lizenz vergeben für Edith -->
  <grant>
    <keyHolder licensePartId="Edith">
      <info> <!-- Digitale Signatur von Edith --> </info>
    </keyHolder>
    <mx:play/>
    <mx:diReference>
      <mx:identifizier>urn:harald-1</mx:identifizier>
    </mx:diReference>
    <validityInterval>
      <notBefore>2004-01-01T00:00:00</notBefore>
      <notAfter>2004-12-31T12:59:59</notAfter>
    </validityInterval>
  </grant>
  <!--Die Lizenz ist von Harald ausgestellt-->
  <issuer>
    <keyHolder>
      <info> <!-- Digitale Signatur von Harald --> </info>
    </keyHolder>
  </issuer>
</license>
```

Auswahlstruktur des Digital Items Das Auswahlstrukturobjekt entspricht dem DIDL Choice-Fragment und umfasst Name, Typ, Frage und Alternativen einer Auswahlstruktur. Der Name identifiziert die Auswahlstruktur innerhalb des Digital Items eindeutig. Der Typ legt die Anzahl an wählbaren

⁵ definiert in XrML (the eXtensible Markup Language), einem allgemeinen Digital Right Language Standard, siehe <http://www.xrml.org/>.

Alternativen. Die Frage zeigt dem Benutzer an, was er in dieser Auswahlstruktur auswählen kann. Die Alternativen einer Auswahlstruktur werden intern durch einen Namen und dem Benutzer gegenüber durch eine Beschreibung repräsentiert. Ein Auswahlstrukturobjekt erzeugt aus seinen Daten ein DIDL Choice-Fragment.

Das folgende XML-Fragment zeigt eine mögliche Auswahlstruktur zwischen zwei Videoauflösungen.

```
<Choice minSelections="1" maxSelections="1">
  <Descriptor>
    <Statement mimeType="text/plain">
      Welche Auflösung bevorzugen Sie?
    </Statement>
  </Descriptor>
  <Selection select_id="MPEG_klein_FORMAT">
    <Descriptor>
      <Statement mimeType="text/plain">Ich gebe mich mit normaler Auflösung zufrieden
    </Statement>
    </Descriptor>
  </Selection>
  <Selection select_id="MPEG_gross_FORMAT">
    <Descriptor>
      <Statement mimeType="text/plain">Ich möchte unbedingt hohe Auflösung</Statement>
    </Descriptor>
  </Selection>
</Choice>
```

Diese Auswahlstruktur ist an die Auswahl der Ressourcen wie folgt geknüpft:

```
<Component>
  <Condition require="MPEG_klein_FORMAT"/>
  <Resource mimeType="video/mpeg" ref="video1.mpeg"/>
</Component>
...
<Component>
  <Condition require="MPEG_gross_FORMAT"/>
  <Resource mimeType="video/mpeg" ref="video2.mpeg"/>
</Component>
```

Das Komponenten-Objekt Ein Komponentenobjekt besteht aus der URL einer Videodatei, einer Bedingung für die Verwendung der Komponente beim Konsum des Digital Items und optionalen Beschreibungen der Videodatei. Eine Komponente wird beim Konsum des Digital Items nur dann verwendet bzw. verarbeitet, wenn die Bedingung der Komponente wahr ist. Eine Bedingung ist die Konjunktion von mehreren internen Namen von Alternativen (siehe Auswahlstrukturobjekt) und ist dann wahr, wenn alle Alternativen, auf die sich die internen Namen der Bedingung beziehen, durch den Benutzer ausgewählt werden. Zu einer Videodatei können optional eine MPEG-7 GoF/GoP-Beschreibung und eine MPEG-21 DIA gBSDL-Beschreibung erstellt werden. Die Beschreibungen der Videodateien werden als eigenständige DIDL Descriptor-Fragmente eingebunden. Das Komponentenobjekt erstellt ein DIDL Component-Fragment, in das es die DIDL Descriptor-Fragmente der Videodateibeschreibungen integriert.

4.2 Ablauf zur Erstellung eines Items

Die Erstellung eines Digital Items umfasst drei Schritte. Im ersten Schritt werden die Objekte der Objektstruktur mit den Daten, die der Benutzer über die graphische Schnittstelle eingegeben hat, initialisiert. Nach diesem Schritt steht die Objektstruktur zur weiteren Verarbeitung bereit. Zur Erstellung des Digital Items aus der Objektstruktur wird jedes Objekt dazu aufgerufen, sein DIDL XML-Fragment zu erstellen. Diese XML Fragmente sind die Bausteine, aus dem das Digital Item generiert wird. Nachdem jedes

Objekt sein DIDL XML-Fragment erstellt hat, können die einzelnen DIDL Fragmente zu einem Digital Item zusammengefügt werden. Außerdem muss noch der Header des Digital Items erzeugt werden. Der Header besteht aus den vom Digital Item und dessen Komponenten verwendeten XML Namespaces, wobei die Namespaces von den Objekten der Objektstruktur abgefragt werden. Nachdem der Header erzeugt wurde, werden der Reihe nach das DIDL Fragment für den Bezeichner des Digital Items, die DIDL Fragmente für die globalen Deskriptoren, das DIDL Fragment für den Lizenzdeskriptor, die DIDL Fragmente für die Auswahlstrukturen und Komponenten zu einem Digital Item zusammengefügt. Das Ergebnis dieses Schrittes ist ein Digital Item in XML Notation. Anzumerken ist in diesem Zusammenhang auch, dass wenn sich das XML Schema für die Digital Item Declaration Language ändern würde, nur der Teil der Objekte geändert werden müsste, der das XML Fragment generiert.

4.3 Der DI Builder

Der DI Builder (siehe Screenshots im Annex) als *signiertes* Applet realisiert, erlaubt die einfache und transparente Erstellung von MPEG-21 Digital Items in einer graphischen Oberfläche. Daneben unterstützt der DI Builder den Transfer per FTP von Digital Items und Mediendateien auf Server wie auch den Download von Mediendateien ins lokale Dateisystem des Benutzers. Der DI Builder kann unter: <http://www-itec.uni-klu.ac.at/~harald/MPEG-21/> getestet werden.

Der DI Builder (siehe Abbildung 5) ist in zwei logische Teile gegliedert, die graphische Oberfläche zur Eingabe der Daten für die Erstellung von Digital Items und den Logikteil, dessen Hauptkomponente die Objektstruktur ist. Die Erstellung eines Digital Items ist in drei Schritte unterteilt. Im ersten Schritt füllt der Benutzer in der graphischen Oberfläche ein Eigenschaftsobjekt mit Eigenschaft-Wert-Paaren. Beispiele für eine Eigenschaft ist die FTP-Adresse an welche die Mediendateien transferiert werden sollen. Im Logikteil wird das Eigenschaftsobjekt geparkt und daraus die Objektstruktur erstellt. Abschließend wird aus der Objektstruktur ein MPEG-21 konformes Digital Item generiert. Die Kommunikation zwischen Oberfläche und Logikteil beschränkt sich auf die Übergabe des Eigenschaftsobjekts. Eine Callback-Funktion wird zur Anzeige des Fortschritts bei der Erstellung des Items in der graphischen Oberfläche benötigt. Durch diese Trennung ist der Logikteil unabhängig vom Implementierungsteil der Oberfläche.

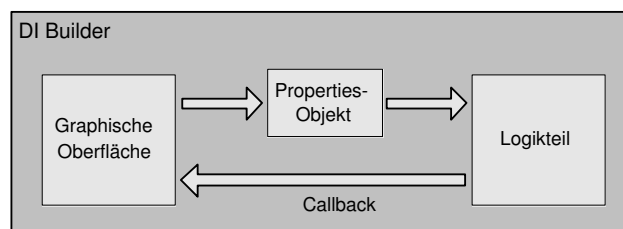


Figure 5. Aufbau der DI Builder Software

5 Methodik und Software zum Konsum von MPEG-21 Digital Items

Die **DI Consumer**-Software erlaubt den Konsum von Digital Items u.a. solche, die mit dem DI Builder erstellt wurden. Der Konsum umfasst dabei die Auswahl von Videodateien des Digital Items, das Abspielen der Videodateien und das Betrachten des Digital Items in einem XML Viewer. Sowohl das Abspielen der Videodateien als auch das Betrachten der Metadaten des Digital Items hängt vom Besitz einer gültigen Lizenz ab, das heißt, nur wenn im Digital Item eine Lizenz mit den notwendigen Rechten integriert ist, darf der Benutzer die Aktionen ausführen. Die Verarbeitung des Digital Items geschieht in so genannten Engines. Die Aufgabe der Engines ist die Verarbeitung von Digital Items mit und ohne Hilfe des Benutzers. Das Ergebnis der Verarbeitung ist abhängig von der jeweiligen Engine und umfasst beispielsweise die Auswahl der Ressourcen aus einem Digital Item oder auch das Überprüfen von Benutzerberechtigungen.

5.1 Methodik zum Konsum von MPEG-21 Digital Items

Der Hauptbestandteil des logischen Teils des DI Consumers sind die *Engines* zur Verarbeitung eines Digital Items. Eine Engine übernimmt ein Digital Item und verarbeitet das Digital Item bzw. Teile des Digital Items mit oder ohne der Hilfe des Benutzers. Der DI Consumer besteht aus drei Engines. Die *ChoiceEngine* verarbeitet die Auswahlstrukturen eines Digital Items mit Hilfe des Benutzers und füllt einen globalen Ressourcenbuffer mit den Videodateien an, die durch die gewählten Alternativen bestimmt werden. Danach wird das Digital Item an die *ProcessingEngine* weitergeleitet. Die *ProcessingEngine* adaptiert die Videodaten auf Wunsch, spielt sie dann in einem Videoplayer ab und zeigt das Digital Item in einem XML Viewer an. Vor der Anzeige eines Videos bzw. des Digital Items ruft die *ProcessingEngine* die *LicenseEngine* auf. Die *LicenseEngine* prüft, ob der Benutzer das Video und das Digital Item ansehen darf. Nur wenn die *LicenseEngine* ein positives Evaluierungsergebnis liefert darf der Benutzer das Video abspielen. Der Fluss eines Digital Items durch die einzelnen Engines ist in Abbildung 6 festgehalten. Im folgenden beschreiben wir die Aufgaben der Engines.

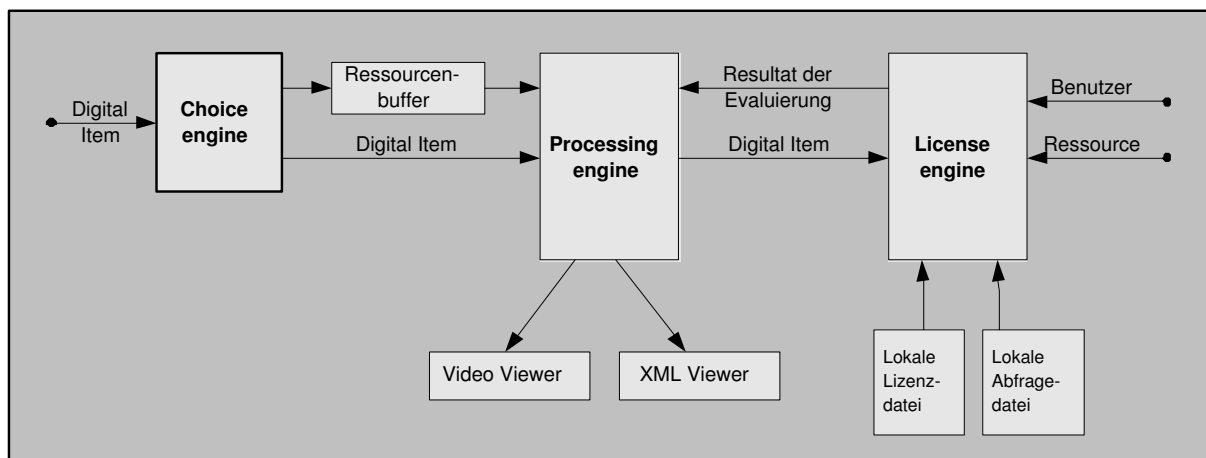


Figure 6. Engines des DI Consumers

Aufbau und Funktionalität der Engines Die DI Consumer-Software (siehe Abbildung 7) besteht aus einer graphischen Oberfläche, Engines zur Verarbeitung von Digital Items und einer zentralen Schnittstellenkomponente zur Interaktion zwischen der graphischen Oberfläche und den Engines des DI Consumers. In der graphischen Oberfläche wird der Konsum von Digital Items durch den Benutzer gesteuert, wobei die Komponenten der graphischen Oberfläche über die Schnittstellenkomponente auf die Engines und deren Funktionalität zugreifen. Die Aufgabe der Engines ist es, ein Digital Item zu übernehmen und zu verarbeiten, wobei der Verarbeitungsschritt von der spezifischen Engine abhängt. Jede Engine implementiert ein Interface mit einer Methode zur Verarbeitung eines Digital Items.

Funktionsweise der ChoiceEngine Die ChoiceEngine verarbeitet die Auswahlstrukturen eines Digital Items mit Hilfe des Benutzers und füllt den globalen Ressourcenbuffer des DI Consumers. Der Ressourcenbuffer enthält nach der Verarbeitung der Auswahlstrukturen alle Mediendateien, die den Auswahlkriterien des Benutzers entsprechen. Die Auswahlstrukturen lassen sich sehr gut in der graphischen Oberfläche darstellen (mit Hilfe von Radiobuttons bzw. Checkboxes). Der Benutzer sieht nur die Frage und die Alternativen der Auswahlstruktur.

Funktionsweise der ProcessingEngine Die ProcessingEngine spielt ein Video ab und stellt ein Digital Item in einem XML Viewer dar. Bevor ein Video durch den Benutzer abgespielt werden darf, wird von der ProcessingEngine die LicenseEngine aufgerufen. Die LicenseEngine prüft, ob der Benutzer zum Abspielen des ausgewählten Mediendatei berechtigt ist. Nur wenn die LicenseEngine ein positives Evaluierungsergebnis liefert, wird die Mediendatei abgespielt und der XML Viewer aktiviert. Die ProcessingEngine integriert auch eine *AdaptationEngine*, welche eine Ressource vor dem tatsächlichen Konsum auf Benutzeranforderung adaptiert. Die AdaptationEngine verwendet die BSD-Description des Digital Items für die Anpassung eines Videos, beispielsweise zum Entfernen von Frames, um die Band-

breitenanforderung zu reduzieren (siehe auch Kapitel 2.2). Der Benutzer kann um eine solche Adaptierung ansuchen, wenn Betriebsmittelengpässe vorliegen.

Funktionsweise der LicenseEngine Die LicenseEngine überprüft, ob der aktuelle Benutzer eine gewählte Mediendatei zum aktuellen Zeitpunkt abspielen darf. Darunter fällt auch die Prüfung, ob der Benutzer die Anzahl erlaubter Wiedergaben noch nicht überschritten hat. Nur wenn die LicenseEngine ein positives Resultat liefert, wird die Mediendatei im DI Consumer wiedergegeben und es dem Benutzer erlaubt, das Digital Item in einem XML Viewer anzusehen.

5.2 Software zum Konsum von MPEG-21 Digital Items

Der *DI Consumer* ermöglicht den Konsum von MPEG-21 Digital Items. Der Konsum eines Digital Items umfasst die Auswahl von Mediendateien, das Abspielen der Mediendateien und das Betrachten der Metadaten des Digital Items. Das Abspielen der Mediendateien und das Betrachten der Metadaten durch den Benutzer hängt davon ab, ob der Benutzer durch die in den Digital Item spezifizierten Lizenzen dazu berechtigt wurde.

Der Konsum eines Digital Items wird über die graphische Oberfläche des DI Consumers gesteuert. Um das zuvor erwähnte Anwendungsszenario zu unterstützen, wurde ein HTTP-Downloader integriert, mit dem Digital Items von öffentlichen Servern heruntergeladen werden können. Der DI Consumer kann unter: <http://www-itec.uni-klu.ac.at/~harald/MPEG-21/> getestet werden.

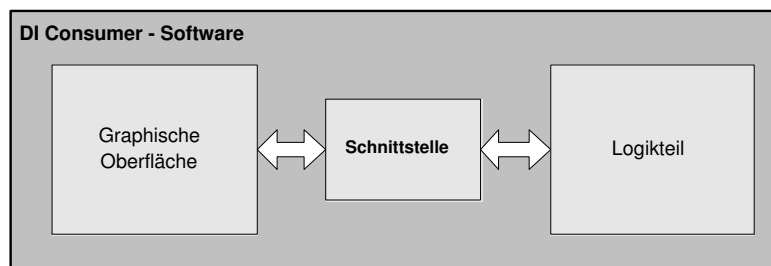


Figure 7. Struktureller Aufbau der DI Consumer-Software

Die DI Consumer-Software (siehe Abbildung 7) besteht aus einer graphischen Oberfläche, Engines zur Verarbeitung von Digital Items und einer zentralen Schnittstellenkomponente zur Interaktion zwischen der graphischen Oberfläche und den Engines des DI Consumers. In der graphischen Oberfläche wird der Konsum von Digital Items durch den Benutzer gesteuert, wobei die Komponenten der graphischen Oberfläche über die Schnittstellenkomponente auf die Engines und deren Funktionalität zugreifen.

6 Schlussfolgerungen

MPEG-21 ist der "Multimedia-Framework" Standard, welcher den Austausch von Multimedia-Daten und Inhalten benutzerfreundlicher, transparenter und fairer gestalten soll. Ein Digital Item ist dabei die grundlegende Entität zur Verteilung von Informationen im Multimedia-Framework. Es ist eine Zusammenfassung von Ressourcen (z.B. ein Video) und Metadaten (z.B. eine Beschreibung der Farbverteilungen im Video). Die ersten Teile von MPEG-21 sind kürzlich veröffentlicht worden, oder werden bis Jahresende zur Verfügung stehen. Werkzeuge zur Erstellung und Verarbeitung von Digital Items, welche dem Framework-Gedanken von MPEG-21 folgen, sind deswegen unerlässlich.

In diesem Zusammenhang stellt unser Artikel eine Methodik und Software zur Erstellung und zum Konsum von MPEG-21 Digital Items zur Verfügung. Die entwickelte Methodik zur Erstellung von Digital Items, insbesondere die Abbildung eines Digital Items in einer Objektstruktur, ist einfach zu handhaben, weiters allgemein und leicht erweiterbar. Die zwei entwickelten Softwarewerkzeuge *DI Builder* und *DI Consumer* ermöglichen einerseits das (transparente) Erstellen und den Konsum von Digital Items für Anwender, und zeigen andererseits eine allgemeine Methodik zur Verarbeitung von Digital Items auf.

Wir planen zur Zeit gerade die Verknüpfung unserer Softwarewerkzeuge mit Caliph and Emir [17]⁶, einem MPEG-7 Bildannotierungswerkzeug, welches es uns ermöglicht, auch für Bilder MPEG-7 Beschreibungen zu generieren.

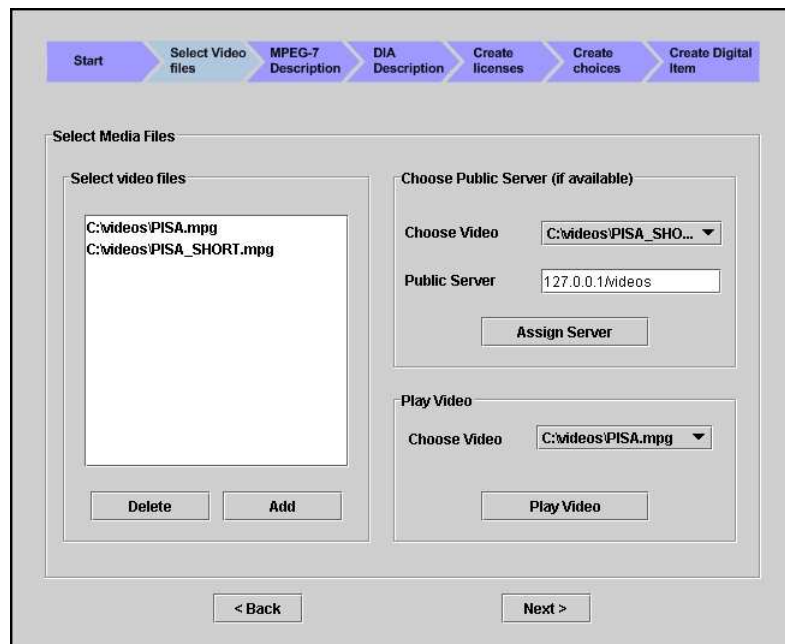
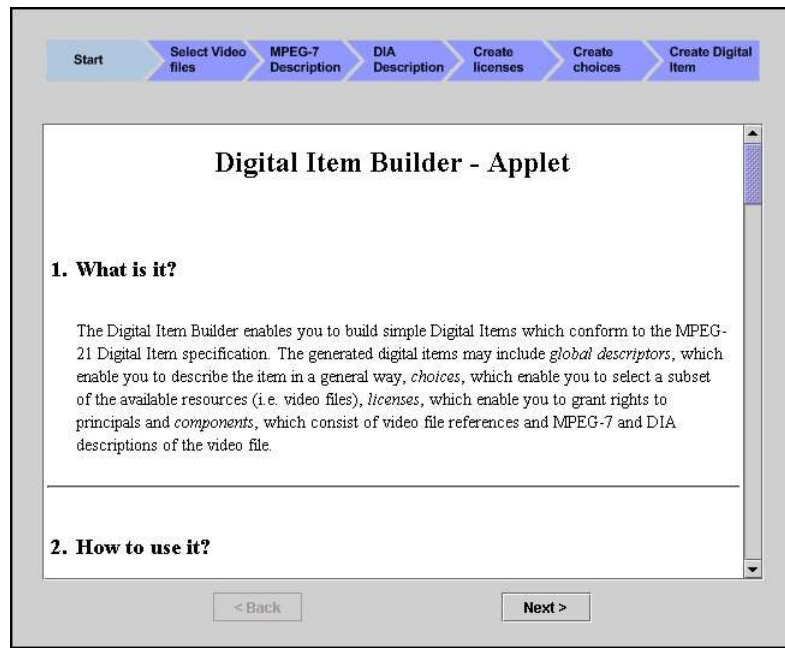
References

1. Burnett, I. Van de Walle, R. Hill, K. Bormans, J. Pereira, F.: "MPEG-21: Goals and Achievements", IEEE MultiMedia 10(4):60-70, Oct-Dec 2003.
2. Watkinson, J.: "MPEG Handbook", Focal Press, 395 pages, Sep 2001. ISBN: 0240516567.
3. Haskell, B. Puri, A. Arun, N.: "Digital Video: An Introduction to MPEG-2", Kluwer Academic Publishers, 441 pages, Dec. 1996. ISBN: 0412084112.
4. Pereira, E. Ebrahimi, T., : "The MPEG-4 Book", Prentice Hall, Jul 2002. ISBN: 0130616214.
5. Kosch, H. Heuer, J.: "MPEG-7", Aktuelles Schlagwort, Informatik Spektrum, 26(2):105-107, April 2003.
6. Pereira, F.: "The MPEG-21 Standard: Why an Open Multimedia Framework?". In the Proceedings of the 8th International Workshop on Interactive Distributed Multimedia Systems (IDMS 2001), LNCS 2158, Springer Verlag, pp. 219-220, Lancaster, UK, September 2001.
7. Dublin Core Metadata Initiative(DCMI): "DCMI Metadata Terms". Verfügbar unter <http://dublincore.org/documents/dcmi-terms/>, Jul 2004.
8. Becker, C. Böszörményi, L. Kosch, H. Sary, C.: "Distributed Multimedia Object/Component Systems and QoS in Distributed Object Systems". In Proceedings of the Workshops held at the 15th European Conference for Object-Oriented Programming (ECOOP) 2001, LNCS 2323, Springer Verlag, Budapest, Hungary.
9. Xu, D. Hefeeda, M. Hambrush, S. Bhargava, B. : "On Peer-to-Peer Media Streaming". In Proceedings of IEEE International Conference on Distributed Computing Systems (ICDCS'02), Vienna, Austria, pp. 363-373, July 2002.
10. Mühlhäuser, M. Gecsei, J.: "Services, Frameworks, and Paradigms for Distributed Multimedia Applications", IEEE MultiMedia, 3(3):48-61, 1996.
11. Rights.com: "The MPEG-21 Rights Expression Language", White Paper der Rightscom Ltd, Jul 2003. Verfügbar unter http://www.rightscom.com/Portals/0/whitepaper_MPEG21-RELCB.pdf.
12. Vetro, A.: "MPEG-21 digital item adaptation: enabling universal multimedia access", IEEE MultiMedia 11(1):84-87, Jan-Mar 2004.
13. Panis, G. Hutter, A. Heuer, J. Hellwagner, H. Kosch, H. Timmerer, C. Devillers, S. Amielh, M.: "Binary Resource Adaptation Using XML Bitstream Syntax Description", EURASIP Signal Processing: Image Communication, 18(8):721-747, September 2003.
14. Sikora, T.: "The MPEG-7 Visual Standard for Content Description-An Overview", IEEE Transactions on Circuits and Systems for Video Technology, 11(6):703-315, June 2001.
15. Kosch, H.: "Distributed Multimedia Database Technologies supported by MPEG-7 and MPEG-21", CRC Press, 248 pages, November 2003. ISBN: 0-849-31854-8.
16. Döller, M. and Kosch, H.: "An MPEG-7 Multimedia Data Cartridge". In Proceedings of the SPIE Conference on Multimedia Computing and Networking 2003 (MMCN03), Santa Clara, CA, January 2003.
17. Lux M., Klieber W., Becker J., Tochtermann K., Mayer H., Neuschmied H., Haas W.: "XML and MPEG-7 for Interactive Annotation and Retrieval Using Semantic Meta-data", Journal of Universal Computer Science, 8(10):965-984, Okt. 2002.

⁶ siehe auch <http://caliph-emir.sourceforge.net/>

Annex: Screenshots des DI Builder

In diesem Anhang sind einige Screenshots des DI Builder in der Reihenfolge ihres Aufrufs angeführt. Der DI Builder besteht aus der Start-Maske, der Select Video Files-Maske, der MPEG-7 Description-Maske, der DIA Description-Maske, der Create Licenses-Maske, der Create Choices-Maske und der Create Digital Item-Maske.



Start > Select Video files > MPEG-7 Description > DIA Description > Create licenses > Create choices > Create Digital Item

Create MPEG-7 Description

Select video files

C:\videos\PISA.mpg
 C:\videos\PISA_SHORT.mpg

Set parameters

Choose video: C:\videos\PISA.m...

H/S/V: 16 4 4

Threshold (in %): 70

Number of frames for scene recognition

2 frames per second
 All frames (computationally intensive)

Store settings

Start > Select Video files > MPEG-7 Description > DIA Description > Create licenses > Create choices > Create Digital Item

Create licenses

Set license name and resource

License name: Lic1

Resource: C:\videos\PISA.mpg

Addressee

All Single person

First name: Some

Last name: One

Number of executions

Number of allowed executions: 3

Set right

Play
 Copy
 Modify

Validity interval

From: 12.04.2004 00:00

Until: 12.04.2004 00:00

Signer

First name: Some

Last name: Signer