An MPEG-21-driven Utility-based Multimedia Adaptation Decision Taking Web Service

Martin Prangl
Dept. of Information Technology
Klagenfurt University
Universitätsstraße 65–67
9020 Klagenfurt, Austria
martin.prangl@itec.uni-klu.ac.at

Ingo Kofler
Dept. of Information Technology
Klagenfurt University
Universitätsstraße 65–67
9020 Klagenfurt, Austria
ingo.kofler@itec.uni-klu.ac.at

Hermann Hellwagner
Dept. of Information Technology
Klagenfurt University
Universitätsstraße 65–67
9020 Klagenfurt, Austria
hermann.hellwagner@itec.uni-klu.ac.at

ABSTRACT

Supporting transparent delivery and convenient use of multimedia content across a wide range of networks and devices is still a challenging task within the multimedia research community; Universal Multimedia Access (UMA) is a vision that has been pursued for quite some time. In multimedia frameworks, content adaptation is the core concept to make progress toward this goal. Most media adaptation engines targeting UMA scale the content w.r.t. terminal capabilities and network resource constraints and do not sufficiently consider end user preferences or even the utility of the adapted content for the user. Based on our previous work and the support of the MPEG-21 framework, we present a transparent solution to provide a content utility-aware adaptation decision for such utility-unaware multimedia frameworks. The idea is to outsource the challenging utility-aware adaptation decision taking task, which takes many factors into consideration and leads to a complex optimization problem. A realistic use case is adopted to show how related external multimedia frameworks can easily integrate and use our proposed adaptation decision taking Web Service.

Categories and Subject Descriptors

H.4.m [**Information Systems**]: Information Systems Applications—*Multimedia*

Keywords

Universal Multimedia Access, Multimedia Adaptation, MPEG-21, Quality of Service, Adaptation Decision Taking, Web Service

Ambi-sys 2008, February 11-14, 2008, Quebec, Canada.

1. INTRODUCTION

The consumption and exchange of multimedia content over the Internet is becoming more and more popular. Public content providers like YouTube[1] provide free access to a huge and rapidly growing set of video content. TV stations begin to offer their channels or content over the Internet as well. In contrast to usual data transfer, e.g., FTP or e-mail, audiovisual content is characterized by high bit rate and burstiness. In addition to its comparatively high demand on available bandwidth, its consumption is further constrained by its sensitivity to delay, jitter, and data loss rate. In case of an on-the-fly consumption (not just download and play) or live consumption of a TV channel, there can occur problems in case of link bandwidth fluctuations or bottlenecks. Although the network research community investigated strategies to provide Quality of Service (QoS) in packet-switched networks [1][2][3], today the Internet still represents a best effort network.

Nowadays, a massive trend can be seen toward the use of wireless technologies. Modern devices like PDAs or mobile phones are connected to the Internet by WLAN or UMTS carriers, enabling comfortable mobile and location independent consumption of content. However, the use of wireless connections is critical in case of such real time multimedia applications. Bandwidth, delay, and data loss rates are varying in dependence of the signal quality which is sensitive to the location of the client as well as to natural influences. Apart from the type of network link, the terminals on the client side may have different capabilities as well, e.g., the spatial resolution of the display or the codec capabilities of the media player.

In order to support a wide range of terminals, independently of their network connection, multimedia content adaptation is seen as a core concept to overcome the heterogeneity of the available terminals and networks. The basic idea is to modify the content in a way that it meets the terminal capabilities and also does not exceed the available network resources. An example for such an adaptation would be the consumption of a 4CIF-sized H.264 content on a PDA. Due to the device's limited display and decoding capabilities, it might be necessary to re-scale the initial video to the resolution of the device and to transcode it to MPEG-2.

The vision of accessing the multimedia content anywhere

[1] http://www.youtube.com

on any device is known as Universal Multimedia Access (UMA) [4] and currently a significant amount of research takes place in this research area. One drawback of the existing adaptive multimedia frameworks, e.g., [5][6], is that they focus rather on overcoming technical limitations than on the user's requirements and preferences. However, the question "How to adapt multimedia data in order to maximize the user's perceived utility?" is of central relevance and is currently not addressed that explicitly in the existing frameworks. As a consequence, not only technical limitations of the involved devices but also the type of the audiovisual content, e.g., its genre, has to be taken into consideration. For example, it would be preferable w.r.t. Universal Multimedia Experience (UME) [7] to adapt an action video in the spatial domain rather than in the temporal domain [8]. As a consequence, the user would get a smaller video window but he/she would still be able to fully enjoy rapid motion in action scenes.

In our previous work [9], we investigated a utility-based adaptive multimedia framework, which additionally takes user impressions as well as user characteristics as input for the adaptation decision taking process. Once all these input values are given, the adaptation decision taking process can be represented as an optimization problem and a final adaptation decision can be derived. In this paper we propose to "outsource" this decision taking process into a dedicated software component that offers its decision taking service via a Web Service interface. This enables an easy and platform-independent integration into existing adaptation frameworks to enrich them with utility awareness.

The remainder of this paper is organized as follows. Section 2 gives an overview of the possible adaptations that can be performed on audiovisual content. The adaptation decision taking problem is introduced in Section 3. A short introduction into MPEG-21 is given as well, which explains the metadata (descriptors) which are required in order to offer a standardized interface to the user for submitting information about the terminal, the available resources, and the requested content. Based on the behavior of related media frameworks, the demand of our adaptation decision taking service is motivated in Section 4. In order to provide easy access to our service, Section 5 is dedicated to design issues and to the interaction between an adaptive media framework and the serving adaptation decision engine. Future improvements of our service are discussed in Section 6. Section 7 finally concludes this work.

2. CONTENT ADAPTATION

The adaptation of multimedia content is required for closing the gap between the format of the offered content and the limitations of both the consumer's terminal and the networks between the terminal and the source of the content, e.g., the streaming server. Nowadays, there exist many types of terminals ranging from mobile devices to full-featured PCs or dedicated set-top boxes. Although this variety of available devices can be seen as a significant advantage for the consumer, this results in technical challenges since all these devices differ significantly in their capabilities from both a hardware and software point of view. Hardware-related capabilities might limit the maximum display resolution and the number of speakers for audio playback. The software, i.e., the player that is finally responsible for decoding the multimedia content, might also come along with restrictions concerning the available decoders. This might either mean that a certain decoder is not installed on that terminal or that the processing power of the device is insufficient for decoding a specific content in real time (which is mostly the case for new codecs like H.264 or its scalable extension). The second reason for doing adaptation is to shape the multimedia content such that it meets the given bandwidth limitations of the network.

Adaptation of audio and video content can be performed in many dimensions which are briefly discussed in the following. Please note that this is not an exhaustive list but includes those adaptation techniques that are commonly used. Another kind of adaptation which we do not consider would be "transmoding", which deals with changing the modality of a multimedia content, e.g., transforming a video clip into a slide show.

2.1 Video Adaptation

The video stream of a movie can be adapted in several ways. *Spatial* adaptation means to change the spatial resolution of each frame of the video. It is possible to achieve a decrease of spatial resolution by cropping out a region of interest (ROI) or by applying spatial resampling techniques. Resampling is usually done by pixel dropping, e.g., dropping every second pixel and every second row, or neighborhood aware pixel replacing algorithms, e.g., median replacing [10].

Temporal video adaptation means to generate a variation of the original video which differs in the number of frames per second (frame rate). A decrease of the frame rate typically causes loss of motion information for the observer. In most of the available video codecs frame dropping can be done in the compressed domain, which means that frames can be dropped from the encoded bit stream. For this reason the adaptation in the temporal domain was often considered as more efficient than spatial adaptation since the processing-intensive decoding and re-encoding steps can be omitted. However, the emerging of scalable video codecs like the scalable extension of H.264 [11] also enables a cheap adaptation in the spatial dimension which will make this advantage more and more irrelevant.

Frame quantization is a codec specific adaptation step in the Signal-to-Noise Ratio (SNR) domain [12]. Video encoding is usually not performed in the temporal domain. The reason is that it is much more efficient to transform the video signals into the frequency domain. This codec specific transformation is usually performed by Fourier, Cosine, or Wavelet transforms. The resulting frequency coefficients have theoretically infinite fidelity. In order to keep the encoding overhead of these coefficients low they get quantized and higher ordered (less important) ones are resulting in a zero value. The granularity of coefficient quantization is given by well defined quantization matrices. In other words, the quantization step introduces a certain degree of information loss. The quantized coefficients are subsequently entropy encoded which profits from the fact that the less important coefficients became zero. The decoder at client side performs the inverse sequence. The critical decoding step forms the re-transformation from the frequency into the temporal domain. Since coefficients were quantized or dropped by the encoder, the decoder suffers from incomplete information and fails in exactly reproducing the original (uncompressed) frame. Since most of the common codecs are based on a Discrete-Cosine Transformation (DCT), that op-

erates on a block of samples, the loss of of coefficients lead to artifacts at the border of each block. As a consequence, the observer perceives a certain degree of blockiness or blurriness in the video frames. Although recent DCT based codecs [13] offer features like a deblocking filter that reduces this effect which increases with the level of quantization.

2.2 Audio Adaptation

Adapting the audio part of a movie is in general less expensive than video adaptation because audio signals require much less bit rate than video signals. Similar to a video stream the audio stream can also be adapted in several dimensions. First, the *sample rate* can be adjusted. The sample rate defines the number of samples which are taken per second of the original audio signal. It strongly relies on Shannon's theorem, which indicates that reducing the sample rate leads to dropping of higher frequencies of the original audio signal. Second, the *number of bits per sample* can be modified, which are used by the quantization process. The lower the average number of bits that are used per sample, the lower the bit rate of the encoded audio stream will be. Finally, also the *number of audio channels* can be reduced, e.g., from Dolby 5.1 to a mono signal.

3. ADAPTATION DECISION TAKING

As depicted above, audiovisual content can be adapted along many dimensions. This leads to the question how to adapt the content in order to meet the given resource limitations and terminal capabilities. The process of finding appropriate adaptation parameters is known as adaptation decision taking and the software component that actually performs this task is consequently called Adaptation Decision Taking Engine (ADTE). In order to take an adaptation decision, the ADTE requires information about the terminal capabilities like supported audio and video codecs, the spatial resolution of the display, the maximum audio frequency of the speakers, and the number of installed speakers. The knowledge about the available bandwidth of the client's network connectivity and the available CPU power of the terminal are important as well. Based on this information about the *usage context* of the content, the task of the ADTE is to determine a content variation that does not exceed the terminal capabilities and resource limitations but uses them in an efficient way. Since the content can be adapted along different dimensions there might exist more than one set of adaptation parameters that would meet the usage context's limitations. For instance, a video could be adapted by reducing its frame rate *or* by modifying the quantization parameter to achieve a certain bit rate. The challenging task is then to find a final adaptation decision from the set of possible adaptations that represents the optimal choice for the consuming user. This issue makes the adaptation decision taking problem interesting and challenging.

3.1 Utility-based Adaptation Decision Taking

Most adaptive multimedia frameworks consider the terminal as the end of the multimedia delivery chain. In our opinion, however, the user who consumes the content is more relevant. Therefore, the decision taking process should try to maximize the utility of the content w.r.t. the user. This leads to a utility-based adaptation decision taking concept which tries to offer the best possible content variation for a specific use case. For example, in case of an action movie it might be better to adapt the video in the spatial domain rather than in the temporal domain in order to maintain the motion information as fully as possible. For the audio part, it might be better to reduce the sample rate rather than decreasing the number of audio channels for maintaining the surround audio effect. In case of a news (head and shoulder) content, a higher spatial resolution might be preferred compared to a higher temporal resolution. One audio channel (mono) might be enough in this case. However, such assumptions have to be taken with care since the users' impressions are subjective which makes modeling of the content's utility for the user more challenging.

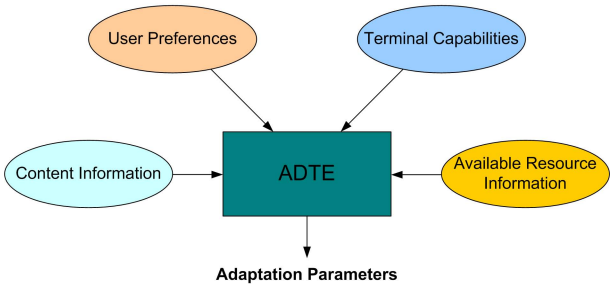

Figure 1: Information for utility-based adaptation decision taking.

In order to offer such a utility-aware adaptation decision, the information about terminal capabilities and the available resources (e.g., network, CPU, memory) alone is insufficient. Additionally, the decision has to take into account the user's individual preferences as well as the information about the requested content, as depicted in Figure 1. All this information is required to find the content variation which is optimal for the specific user request. The resulting adaptation decision is expressed by adaptation parameters, consisting of the elementary stream features of the audio and video part.

The core component of a utility-aware ADTE is a so called *utility model*. The aim of the utility model is to map a certain media stream variation to a certain scalar utility value under consideration of the influences discussed above. In case of audiovisual content, a so called cross-modal utility model [14] is applied, which additionally takes cross-modal influences between the audio and the video parts into consideration for determining the overall utility value of a degraded audiovisual variation.

In general, cross-modal utility models rely on the combination of the elementary stream utilities as shown in Figure 2. First, the (degraded) movie is demultiplexed into its elementary audio and video streams. Then, the utility analysis is performed on both of them separately. The determined audio and video utility values (Ua and Uv) form the input for the cross-modal utility calculation. Its output forms the global audiovisual utility U.

3.2 MPEG-21 for Standardized Context Information Exchange

As the ADTE requires the information of the usage context, it is necessary to communicate it to the ADTE. In order to achieve a high degree of interoperability and to support a wide range of devices, it is desirable to use standardized descriptions.

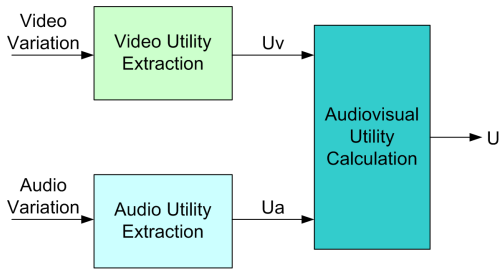

Figure 2: Cross-modal utility extraction.

The MPEG-21 Digital Item Adaptation (DIA) standard [15] aims to support the realization of UMA, which means to provide standardized tools for enabling transparent and augmented use of multimedia content across a wide range of networks, devices, and user preferences. The main idea of MPEG-21 is the exchange of so called Digital Items (DIs) on the delivery chain between the content provider (the media server) and the consumer. A DI represents a structured digital object, consisting of the media content itself as well as of its corresponding metadata. DIs can be defined using normative XML syntax called Digital Item Declaration (DID). In alignment with the UMA vision, the adaptation of such DIs may be necessary somewhere in the delivery chain. According to that circumstance, MPEG-21 provides a set of descriptors (called *tools*) for steering such adaptations. In the following, two tools that are relevant for our approach are briefly discussed.

The Digital Item Declaration (DID) contains content specific information. This includes syntactic information of the media stream (the elementary stream features) like spatial resolution, frame rate etc., as well as semantic information like title, publisher, actors, summaries and so on. Note that the syntactic as well as the semantic media information are provided by MPEG-7 descriptors, which are embedded within the DID. Furthermore, the DID contains information about the location of the content to indicate how the content can be accessed. This information is commonly provided by a Universal Resource Identifier (URI).

The Usage Environment Description (UED) tool allows to specify information about the user, his/her terminal, the network connection as well as the natural environment in which the Digital Item is finally to be consumed. Important examples are the display resolution or the number of speakers which are supported by the terminal and the available bandwidth of the link to which the terminal is connected. Another example that relates to the natural environment is the brightness and loudness at the client's location.

A typical MPEG-21-based client-server interaction is shown in Figure 3. The client initiates a content request to the adaptive media server which includes an instance of the UED containing the necessary terminal, network, and user-specific information. The transport of the UED is not normative. One possibility is to include it in the HTTP request. The media server takes some adaptation decision (utility-aware or not) and adapts the media content according to the decision. Together with its associated metadata (DID), the adapted media content is delivered to the requester in form of a DI.

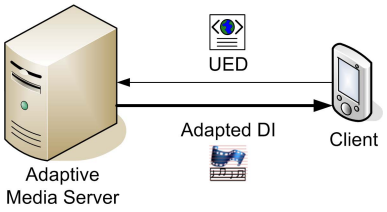

Figure 3: MPEG-21-based client-server interaction.

4. NEED FOR A UTILITY-BASED ADAPTATION DECISION TAKING SERVICE

Multimedia content adaptation decision taking eventually leads to a complex optimization problem. For this reason, many adaptive multimedia frameworks consider only a subset of the proposed influences. Two state-of-the-art systems are discussed w.r.t. this issue in the following.

4.1 The koMMa Framework

The koMMa framework (knowledge-based multimedia adaptation) [5] addresses the UMA problem as a multi-step adaptation process composed of simple adaptation operations to be performed in sequence. The decision which elementary steps are to be applied in which order for content adaptation, is performed by solving a planning problem which is well known within the Artificial Intelligence community. The start state of the proposed planning problem is defined by the initial stream features, e.g., codec, spatial resolution etc. The goal state represents the client's UED, in which the terminal's capabilities, e.g., supported codec, spatial resolution, etc. are defined. A simple example would be a video clip that is initially encoded in MPEG-2 using a spatial resolution of 720x576. In the UED, the codec capability and the spatial display resolution of the consuming device is defined by MPEG-1 and 320x200. In order to perform this adaptation step, koMMa tries to concatenate simple adaptors in a way that the target content variation is produced. Every adaptor performs an elementary adaptation step in this planned sequence. In case of this simple example, it would compose an adaptation chain consisting of an MPEG-2 decoder, a YUV spatial downscale adaptor, and an MPEG-1 encoder. Since this approach is based on a simple planner that can handle only a single goal state, it is assumed that the UED is unambiguous and precisely expresses the user's requirements. That is, the properties defined in the UED specify a single targeted content variation. The implementation of koMMa leads to conflicts w.r.t. UMA in its general sense as explained in the following.

If the UED contained the user's "precise requirements", every user could define the best possible stream features which the terminal would be able to deal with, e.g., by specifying the maximum display resolution, the highest frame rate and bit rate. As a consequence, koMMa would try to perform an adaptation according to these "wishes" at the given resource limitation (network bandwidth). In many situations, such an adaptation will not be possible. For example, it is not possible to encode a video at 720x576 pixels with 25 frames per second at 150 kbit/s, even if the terminal's capabilities indicate that the terminal is able to render it. In such a case, koMMa would not consider a lower spatial resolution for the adaptation in order to overcome the given

resource limitation. As a consequence, the framework fails to ensure UMA. Furthermore, even if the terminal would support an additional, more efficient codec which were able to achieve the given resource limit, this codec would not be considered. The reason is that koMMA is not able to consider an ambiguous UED, e.g., a UED that contains more than one supported codec. In such a case one corresponding element of the UED (codec type) is selected arbitrarily. As a concluding fact, koMMA is a non-utility aware multimedia framework, which does definitely not consider more possible content variations which fit the given resource limitations.

4.2 The CAIN Framework

A very similar multimedia framework presented in [6] is called CAIN. Similar to koMMa, a set of well described adaptation tools, so called Content Adaptation Tools (CATs), are subsequently applied in series, representing the adaptation chain for the given use case. In contrast to koMMa, CAIN accepts two UEDs from the client, one mandatory UED which contains the terminal capabilities which have to be fulfilled and one desirable UED which contains the user preferences, representing constraints which should be fulfilled (e.g., frame rate $>$ 10 fps). The desirable UED helps the ADTE to decide which possible content variation is better. The adaptation decision taking algorithm of CAIN relies on a constraint matching problem where constraints are given by the resource limitations as well as the terminal capabilities. If there are more possibilities (adaptation strategies) to match these constraints, the one which matches the most constraints of the desirable UED is selected.

The assumption of this approach is that the user knows his/her preferences for the given content in advance and that the corresponding desirable UED is provided. Furthermore, CAIN's adaptation decision taking algorithm does not consider cross-modal influences for finding the "best" adaptation for the consumer.

4.3 Discussion and Consequence

The main concern about the discussed frameworks is the assumption that the user knows his/her own preferences in advance (before the content consumption). Generally, users do not know their favorite stream features (spatial resolution, frame rate, sample rate, etc.) in a specific media consumption scenario, since content type, device properties, and network connectivity span a space that is too complex to be explored by a human.

As a consequence, the users are in general not able to provide a reliable UED that would represent their preferences or requirements precisely enough. For this reason, we propose a Utility-based Adaptation Decision Taking (ADT) Service, which is accessible to such non-utility aware frameworks with ease and without major design and implementation changes.

5. DESIGN OF A UTILITY-BASED ADAPTATION DECISION TAKING SERVICE

In our previous work [9], we presented a utility-based adaptive multimedia framework. It captures information about the user, his/her environment, the terminal and the requested content in order to decide which content variation will presumably be the best for the requesting user with respect to his/her utility aspects. Our adaptive multimedia framework continuously learns about the individual user's preferences, which is achieved by collecting feedback ratings after content consumption. This behavior enables the framework to perform reliable adaptation decisions for individual use cases.

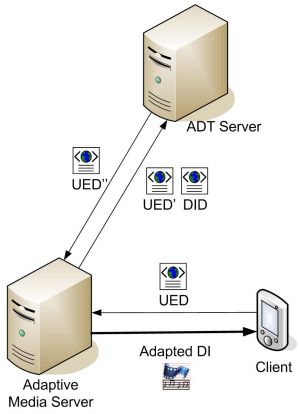

Figure 4: Using an ADT Server for requesting a utility-based adaptation decision.

5.1 Integration of the ADT Service

The concept of a public Adaptation Decision Taking Service that provides utility awareness for already existing multimedia frameworks, is shown in Figure 4. The user at the client side requests a specific content and submits his/her UED to the media server. This UED describes the terminal capabilities as well as the user characteristics. Please note that from the client's point of view there is no difference to the simple use case as discussed before. This indicates that no changes have to be applied at the client side. A fragment of a UED that describes the user characteristics is given in Figure 5. It includes information about the user's citizenship (described by the *PersonType*) together with other personal data like name and email address. The user's favorite genres are described by the *ClassificationPreferences* descriptor and the auditory impairment is described using the *AuditoryImpairment* type. The impairment of a particular ear (the attenuation at a given frequency) is specified in decibel (dB). An example how a UED can be used to describe the terminal itself is given in Figure 6. The necessary information about each terminal consists of its type (PC, notebook, PDA, or mobile), the spatial resolution of the display, the available codecs, the sound specific properties like maximum speaker frequency and the number of speakers. The example describes a PDA with a display of 320x200 pixels and two speakers which are able to handle frequencies between 30 Hz and 18 kHz. This PDA is able to decode MPEG-1 and MPEG-2 video as well as mp2 and mp3 audio.

The integration of the Adaptation Decision Taking Service modifies the initial request handling at the adaptive media server as follows. Once the server receives the request with the UED, it simply forwards the UED to the ADT Server instead of adapting the requested content only considering the terminal capabilities. If the media sever supports only a limited set of audio and video codecs, it has to filter out the unknown ones from the original UED before forwarding it. The forwarded (and potentially filtered) UED is denoted as UED' in Figure 4. In addition to the UED', the ADT server

```
<UsageEnvironmentProperty xsi:type="UsersType">
    <User>
      <UserCharacteristic xsi:type="UserInfoType">
        <UserInfo xsi:type="mpeg7:PersonType">
          <mpeg7:Name>
            <mpeg7:GivenName>Max</mpeg7:GivenName>
            <mpeg7:FamilyName>Mustermann</mpeg7:FamilyName>
          </mpeg7:Name>
          <mpeg7:Citizenship>AT</mpeg7:Citizenship>
          <mpeg7:ElectronicAddress>
            <mpeg7:Email>Max@company.com</mpeg7:Email>
          </mpeg7:ElectronicAddress>
        </UserInfo>
      </UserCharacteristic>

      <UserCharacteristic xsi:type="UsagePreferencesType">
        <UsagePreferences>
          <mpeg7:FilteringAndSearchPreferences>
            <mpeg7:ClassificationPreferences>
              <mpeg7:Genre>
           <mpeg7:Name>Action</mpeg7:Name>
         </mpeg7:Genre>
         <mpeg7:Genre>
           <mpeg7:Name>Cartoon</mpeg7:Name>
         </mpeg7:Genre>
         <mpeg7:Genre>
           <mpeg7:Name>Entertainment</mpeg7:Name>
         </mpeg7:Genre>
            </mpeg7:ClassificationPreferences>
          </mpeg7:FilteringAndSearchPreferences>
        </UsagePreferences>
      </UserCharacteristic>

      <UserCharacteristic xsi:type="AuditoryImpairmentType">
        <RightEar>
          <Freq22kHz>18.0</Freq22kHz>
          <Freq20kHz>14.0</Freq20kHz>
          <Freq18kHz>12.0</Freq18kHz>
          <Freq16kHz>6.0</Freq16kHz>
          <Freq14kHz>8.0</Freq14kHz>
          <Freq12kHz>3.0</Freq12kHz>
          <Freq10kHz>0.0</Freq10kHz>
          <Freq8kHz>0.0</Freq8kHz>
          <Freq6kHz>0.0</Freq6kHz>
        </RightEar>
        <LeftEar>
          <Freq22kHz>57.0</Freq22kHz>
          <Freq20kHz>53.0</Freq20kHz>
          <Freq18kHz>23.0</Freq18kHz>
          <Freq16kHz>9.0</Freq16kHz>
          <Freq14kHz>5.0</Freq14kHz>
          <Freq12kHz>0.0</Freq12kHz>
          <Freq10kHz>0.0</Freq10kHz>
          <Freq8kHz>0.0</Freq8kHz>
          <Freq6kHz>0.0</Freq6kHz>
        </LeftEar>
      </UserCharacteristic>
    </User>
  </UsageEnvironmentProperty>
```

Figure 5: User-specific UED.

```
<UsageEnvironmentProperty xsi:type="TerminalsType">
 <Terminal>
  <TerminalCapability xsi:type="DeviceClassType">
   <DeviceClass href="urn:mpeg:mpeg21:2003:
    01-DIA-DeviceClassCS-NS:2">
    <mpeg7:Name xml:lang="en">PDA</mpeg7:Name>
   </DeviceClass>

  </TerminalCapability>
  <TerminalCapability xsi:type="CodecCapabilitiesType">
   <Decoding xsi:type="AudioCapabilitiesType">
    <Format href="urn:mpeg:mpeg7:cs:
     AudioCodingFormatCS:2001:4.4">
     <mpeg7:Name xml:lang="en">MP3</mpeg7:Name>
    </Format>
   </Decoding>
   <Decoding xsi:type="AudioCapabilitiesType">
    <Format href="urn:mpeg:mpeg7:cs:
     AudioCodingFormatCS:2001:3.2">
     <mpeg7:Name xml:lang="en">MP2</mpeg7:Name>
    </Format>
   </Decoding>

   <Decoding xsi:type="VideoCapabilitiesType">
    <Format href="urn:mpeg:mpeg7:cs:
     VisualCodingFormatCS:2001:2">
     <mpeg7:Name xml:lang="en">MPEG-2</mpeg7:Name>
    </Format>
   </Decoding>
   <Decoding xsi:type="VideoCapabilitiesType">
    <Format href="urn:mpeg:mpeg7:cs:
     VisualCodingFormatCS:2001:1">
     <mpeg7:Name xml:lang="en">MPEG-1</mpeg7:Name>
    </Format>
   </Decoding>

  </TerminalCapability>
  <TerminalCapability xsi:type="DisplaysType">
   <Display id="primary_display">
    <DisplayCapability xsi:type="DisplayCapabilityType"
     bitsPerPixel="24">
     <Mode>
      <Resolution horizontal="320" vertical="200"/>
     </Mode>
    </DisplayCapability>
   </Display>
  </TerminalCapability>

  <TerminalCapability xsi:type="AudioOutputsType">
   <AudioOutput xsi:type="AudioOutputType">
    <AudioOutputCapability
      xsi:type="AudioOutputCapabilitiesType"
      lowFrequency="30" highFrequency="18000"
      numChannels="2"/>
   </AudioOutput>
  </TerminalCapability>
 </Terminal>
</UsageEnvironmentProperty>
```

Figure 6: Terminal-specific UED.

needs information about the requested content. Therefore, the media server sends a DID, describing syntactic as well as some semantic content aspects. The syntactic part provides important information about the encoding of the original audio and video stream, e.g., codec, bit rate, frame rate, spatial resolution, sample rate, etc. An example is given in Figure 7. The semantic part provides general information (summary, actors, etc.) about the content. However, for the adaptation decision only the title and the type of content (genre) is considered.

5.2 Adaptation Decision

Once the ADT server receives the UED' and the DID, it starts its decision taking process. Its task is to select the optimum adaptation parameters according to the adaptive media server's request such that the given bandwidth limit is not exceeded, the terminal capabilities (available codecs, spatial resolution, etc.) are fulfilled, and the user's experience is maximized.

In order to achieve this goal, a hybrid recommender system [16], consisting of a collaborative, a demographic and a knowledge-based engine is invoked to configure our cross-modal utility model [17]. A top level view of our adaptation decision taking approach is given in Figure 8. The recommender takes into consideration both the content information (title, genre) as well as user characteristics. The considered user information includes the citizenship, favorite genres as well as the user's hearing impairments. Based on this

```
<MediaInformation id="S19.2E-0-12480-899">
 <MediaProfile><MediaFormat>
  <Content href="MPEG7ContentCS:2001">
   <Name>audiovisual</Name>
  </Content>
  <BitRate
   average="3500000"maximum="4000000">3500000
  </BitRate>
  <VisualCoding>
   <Format href="urn:mpeg:mpeg7:cs:
    VisualCodingFormatCS:2001:2"
    colorDomain="color">
    <Name xml:lang="en">MPEG-2 Video</Name>
   </Format>
   <Frame height="576" width="720" rate="25.0"/>
  </VisualCoding>
  <AudioCoding>
   <Format href="urn:mpeg:mpeg7:cs:
    AudioCodingFormatCS:2001:3.2">
    <Name xml:lang="en">
     MPEG-1 Audio Layer II</Name>
   </Format>
   <AudioChannels front="0" side="2"
      rear="0" lfe="0" track="0">2
   </AudioChannels>
   <Sample rate="48000.0" bitsPer="4"/>
  </AudioCoding>
 </MediaFormat></MediaProfile>
</MediaInformation>
```

Figure 7: DID describing syntax of original content.

information and by the help of user feedback, maintained in a database, the hybrid recommender calculates the open parameters for our utility model.

The configured utility model is used by the ADT server which is calculating the optimum adaptation parameters based on the border scan optimization algorithm presented in [18]. Its aim is to find the best possible audiovisual content variation (a combination of a certain degraded audio and a degraded video variation) efficiently by only considering those combinations the resource needs of which are under the given constraints. For each possible variation, the corresponding utility value is requested from the utility model. The audiovisual content variation that achieves the highest utility value is finally taken as the result of the adaptation decision taking process. The additional computational cost that is introduced by the decision taking is very low. Typically the optimal adaptation parameters can be calculated in less than 1 second.

Usually, the final adaptation decision is expressed by the stream parameters (frame rate, resolution, etc.) of the best content variation. In case of a public (Web) service, those stream parameters have to be encapsulated in a standardized format. In our case, the adaptation decision is embedded into a UED (further denoted as UED") for keeping transparency for the MPEG-21-aware adaptive media servers that utilize the service. In this case, the media server's task does not differ from the non-utility-aware case, where the UED is coming directly from the user. Figure 9 shows such a UED" corresponding to our previous example. In contrast to the original UED (Figure 6), it contains only one possible codec format for video and audio. Additionally, the average target bit rate information is given for the encoding process (the video bit rate is given by the total bit rate minus the bit rate of the audio stream).

The MPEG-21 DIA standard does not allow to directly define the supported video frame rate of the terminal within a UED. For this reason, the *FillRate* element can be used. The fill rate is defined as the product of frame rate and the horizontal and vertical resolution and is measured in pixels per second. This information enables the adaptive media server to simply calculate the target frame rate by using the signaled fill rate and the requested video resolution. Please note that the user-specific UED is not required for a non-utility-aware media framework. Therefore, it can be simply omitted in UED".

5.3 Metadata Transport

To support an easy integration of the decision taking service into existing frameworks, we propose the usage of Web Service technologies. The widespread SOAP protocol [19] can be used to build services that offer their operations as a kind of remote procedure calls (RPC) in a platform and programming language independent way. This is ensured by the SOAP protocol by exchanging XML-based messages and encapsulating them in HTTP requests and responses. The fact that SOAP is based on XML makes it an ideal candidate for exchanging XML-based metadata like UED and DID, since the protocol offers features like the validation of the included descriptions against their schema definitions. This leads to a more robust communication since invalid descriptions can be detected automatically by the SOAP implementation. The interface of such a Web Service can be defined by using the Web Services Description Language

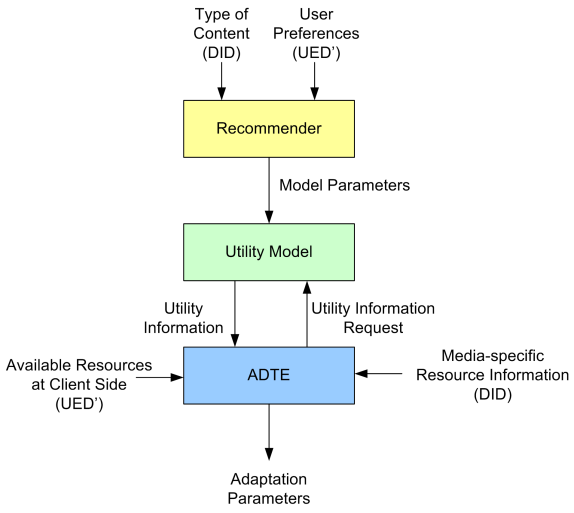

Figure 8: Overview of the Adaptation Decision Taking Process.

(WSDL) [20] which is also based on XML and basically describes all operations offered by the Web Service. Similar to interface descriptions that are written using the Interface Definition Language (IDL), WSDL descriptions can be used to automatically generate wrapper code that makes the invocation of a remote Web Service transparent for the application programmer.

```
<Description xsi:type="UsageEnvironmentType">
 <UsageEnvironmentProperty xsi:type="TerminalsType">
  <Terminal>
   <TerminalCapability xsi:type="CodecCapabilitiesType">
    <Decoding xsi:type="AudioCapabilitiesType">
     <Format href="urn:mpeg:mpeg7:cs:
      AudioCodingFormatCS:2001:4.4">
      <mpeg7:Name xml:lang="en">MP3</mpeg7:Name>
     </Format>
     <CodecParameter xsi:type="CodecParameterBitRateType">
      <BitRate>64000</BitRate>
     </CodecParameter>
    </Decoding>
    <Decoding xsi:type="VideoCapabilitiesType">
     <Format href="urn:mpeg:mpeg7:cs:
      VisualCodingFormatCS:2001:2">
      <mpeg7:Name xml:lang="en">MPEG-2</mpeg7:Name>
     </Format>
     <CodecParameter xsi:type="CodecParameterFillRateType">
      <FillRate>1280000</FillRate>
     </CodecParameter>
     <CodecParameter xsi:type="CodecParameterBitRateType">
      <BitRate>236000</BitRate>
     </CodecParameter>
    </Decoding>

   </TerminalCapability>
   <TerminalCapability xsi:type="DisplaysType">
    <Display id="primary_display">
     <DisplayCapability xsi:type="DisplayCapabilityType">
      <Mode>
       <Resolution horizontal="320" vertical="200"/>
      </Mode>
     </DisplayCapability>
    </Display>
   </TerminalCapability>

   <TerminalCapability xsi:type="AudioOutputsType">
    <AudioOutput xsi:type="AudioOutputType">
     <AudioOutputCapability
      xsi:type="AudioOutputCapabilitiesType"
      lowFrequency="30" highFrequency="18000"
      numChannels="1"/>
    </AudioOutput>
   </TerminalCapability>
  </Terminal>
 </UsageEnvironmentProperty>
</Description>
```

Figure 9: Resulting UED" for non-utility aware adaptive multimedia frameworks.

6. FUTURE WORK

Although our proposed adaptation decision taking component is very mature (as evaluated in [9]) we still identified some possible improvements to the Web Service that will be tackled in the future. Currently, we use the genre and the title of the requested Digital Item to identify the content and to associate the subjective utility in the decision taking process. However, the MPEG-21 standard offers a normative way how to identify a Digital Item. This part of MPEG-21 is called Digital Item Identification (DII) [21] and could improve the ADT Web Service by making it more standard compliant. Another drawback of our current solution is the way how the personal data is handled by the system. Currently, personal data about the preferences of the user, his/her impairments and also the email address is submitted by the adaptive multimedia framework to the Web Service. This can be considered a lack of privacy and for a final commercial solution more research has to be done in order to guarantee the anonymity of the user. In contrast to our previous work [16] this ADT Web Service does not consider the feedback of the user. This was a deliberate design decision since the integration of the feedback loop would impose implications on the available frameworks and would reduce the ease of integration. However, one could investigate how to integrate an optional feedback channel that would require only minor changes to the available frameworks.

7. CONCLUSIONS

In this paper we propose an MPEG-21-driven Utility-based Multimedia Adaptation Decision Taking Web Service, which can be used to enrich existing multimedia adaptation frameworks by providing content utility-aware adaptation decisions. The Web Service offers its functionality through a SOAP interface in order to achieve high interoperability with legacy systems. In addition to that, the use of normative MPEG-21 and MPEG-7 metadata for describing both the usage context and the content itself further guarantees a low integration overhead into existing frameworks.

8. ACKNOWLEDGMENTS

Parts of this work were supported by the EC in the context of the ENTHRONE project (IST-1-507637). Further information is available at http://www.ist-enthrone.org.

9. REFERENCES

[1] D. Estrin, S. Berson, S. Herzog, and D. Zappala. The Design of the RSVP Protocol. Technical report, University of Southern California, Information Sciences Institute, July 1996.

[2] V. Fineberg. A Practical Architecture for Implementing End-to-End QoS in an IP Network. *IEEE Communications Magazine*, 40(1):122–130, January 2002.

[3] X. Xiao and L.M. Ni . Internet QoS: A Big Picture. *IEEE Network*, 13(2):8–18, March/April 1999.

[4] A. Vetro, C. Christopoulos and T. Ebrahami. Special Issue on Universal Multimedia Access. *IEEE Signal Processing Magazine*, 20(2), March 2003.

[5] D. Jannach, K. Leopold, C. Timmerer, and H. Hellwagner. A knowledge-based framework for multimedia adaptation. *Applied Intelligence*, 24(2):109–125, April 2006.

[6] F. López, J.M. Martínez, and V. Valdés. Multimedia Content Adaptation within the CAIN Framework via Constraints Satisfaction and Optimization. In *Proc. 4th Int'l. Workshop on Adaptive Multimedia Retrieval (AMR)*, pages 149–163. Springer LNCS 4398, Geneva, Switzerland, July 2007.

[7] F. Pereira and I. Burnett. Universal Multimedia Experiences for Tomorrow. *IEEE Signal Process. Mag.*, 20(2):63–73, March 2003.

[8] H. Knoche, J.D. McCarthy, and M.A. Sasse. Can Small Be Beautiful? Assessing Image Resolution Requirements for Mobile TV. In *Proc. ACM Multimedia*, pages 829–838, November 2005.

[9] M. Prangl, T. Szkaliczki, and H. Hellwagner. A Framework for Utility-based Multimedia Adaptation. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(6):719–728, June 2007.

[10] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis and Machine Vision*. Thomson-Engineering, 2nd edition, 1999.

[11] ISO/IEC 14496-10:2005/Amd.3:2007. Information Technology – Coding of audio-visual objects - Part 10: Advanced Video Coding, Amendment 3: Scalable Video Coding. 2007.

[12] K.N. Ngan, T. Meier, and D. Chai. *Advanced Video Coding: Principles and Techniques*. Elsevier, 1999.

[13] Th. Wiegand, G.J. Sullivan, G. Bjøntegaard, and A. Luthra. Overview of the H.264/AVC Video Coding Standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7):560–576, July 2003.

[14] D.S. Hands. A Basic Multimedia Quality Model. *IEEE Transactions on Multimedia*, 6(6):806–816, December 2004.

[15] I. S. Burnett, F. Pereira, R. Van de Walle, and R. Koenen, Eds. *The MPEG-21 Book*. Wiley, 2006.

[16] M. Prangl, R. Bachlechner, and H. Hellwagner. A Hybrid Recommender Strategy for Personalized Utility-based Cross-modal Multimedia Adaptation. In *Proc. IEEE Int'l. Conf. on Multimedia and Expo (ICME)*, pages 1707–1710, July 2007.

[17] M. Prangl, H. Hellwagner, and T. Szkaliczki. A Semantic-based Multi-modal Utility Approach For Multimedia Adaptation. In *Proc. 7th Int'l. Workshop on Image Analysis for Multimedia Services (WIAMIS)*, pages 67–70, April 2006.

[18] M. Prangl, H. Hellwagner, and T. Szkaliczki. Fast Adaptation Decision Taking for Cross-modal Multimedia Content Adaptation. In *Proc. IEEE Int'l. Conf. on Multimedia and Expo (ICME)*, pages 137–140, July 2006.

[19] SOAP Version 1.2. W3C Recommendation, 27 April, 2007. URL: http://www.w3.org/TR/soap.

[20] Web Services Description Language (WSDL) 1.1. W3C Note, 15 March 2001. URL: http://www.w3.org/TR/wsdl.

[21] ISO/IEC 21000-3:2003. Information Technology – Multimedia Framework (MPEG-21) - Part 3: Digital Item Identification. 2003.