

Real-Time System for Adaptive Video Streaming Based on SVC

Mathias Wien, *Member, IEEE*, Renaud Cazoulat, Andreas Graffunder, Andreas Hutter, and Peter Amon

(Invited Paper)

Abstract—This paper presents the integration of Scalable Video Coding (SVC) into a generic platform for multimedia adaptation. The platform provides a full MPEG-21 chain including server, adaptation nodes, and clients. An efficient adaptation framework using SVC and MPEG-21 Digital Item Adaptation (DIA) is integrated and it is shown that SVC can seamlessly be adapted using DIA. For protection of packet losses in an error prone environment an unequal erasure protection scheme for SVC is provided. The platform includes a real-time SVC encoder capable of encoding CIF video with a QCIF base layer and fine grain scalable quality refinement at 12.5 fps on off-the-shelf high-end PCs. The reported quality degradation due to the optimization of the encoding algorithm is below 0.6 dB for the tested sequences.

Index Terms—Digital item adaptation, MPEG-21, Scalable Video Coding (SVC), unequal erasure protection (UXP).

I. INTRODUCTION

THE combination of adaptation technology and scalable media formats like Scalable Video Coding (SVC) is about to become applicable for a variety of use cases [9]. Tools enabling the adaptation, e.g., MPEG-21 Digital Item Adaptation (DIA), are already standardized [10], and the specification of SVC as an extension to H.264/AVC will be completed in mid-2007. In order to make the proof-of-concept for the applicability of the technology already during the standardization phase, a real-time system including encoding, error protection, adaptation and decoding was implemented in the DANAE project [5].

SVC compression and adaptation technology was developed for a variety of usage scenarios [2], [3]. These include video broadcast/unicast, video conferencing and surveillance, which are described in more detail in the following.

Manuscript received October 9, 2006; revised June 18, 2006. This work was supported in part by the IST European co-funded project DANAE under Contract IST-1-507113 and in part by Deutsche Telekom Laboratories. This paper was recommended by Guest Editor T. Wiegand.

M. Wien is with the Institut für Nachrichtentechnik, RWTH Aachen University, 52056 Aachen, Germany (e-mail: wien@ient.rwth-aachen.de).

R. Cazoulat is with France Telecom, BP 91226 35512 Cesson-Sevigne, France (e-mail: renaud.cazoulat@orange-ft.com).

A. Graffunder is with T-Systems Enterprise Services, Systems Integration, 10589 Berlin, Germany (e-mail: andreas.graffunder@t-systems.com).

A. Hutter and P. Amon are with Siemens Corporate Technology, Information and Communications, 81730 Munich, Germany (e-mail: andreas.hutter@siemens.com; p.amon@siemens.com).

Digital Object Identifier 10.1109/TCSVT.2007.905519

Digital video broadcast (DVB), IPTV, and video on demand (VoD) are currently deployed solutions for the transmission of video content to one specific user (VoD) or many users (DVB, IPTV). The same content is provided to different end-terminals over various transmission channels at the same time (broadcast/multicast) or at different time instances (unicast). Currently, either the transmission channels or the terminals must be fixed for a service or multiple differently encoded versions of the same content have to be generated and possibly stored. Using scalable video coding as enabled by SVC, a single stream can be used to serve all end-users. Adaptation can be performed at the server but also in the network (e.g., at media gateways) in order to tailor the video stream according to the specific usage requirements.

An advanced adaptation scenario for video conferencing is given by a setup comprising multiple terminals (e.g., PCs, laptops, PDAs, mobile phones) with varying terminal capabilities that are connected via different networks (e.g., fixed-line, mobile) in a joint video conferencing session. For each individual client, the video stream has to be adapted according to its terminal capabilities and connection conditions. In current solutions, transcoders are deployed for this task. If the number of clients is high, transcoding becomes extremely inefficient, since it is computationally complex and also incurs a loss in compression efficiency. Using SVC and adaptation techniques like MPEG-21 DIA, the customization of the video stream to the characteristics of each client (device capabilities, network capacity, user preferences) is facilitated [14]. This yields less expensive hardware components in the network, e.g., in media gateways, and therefore results in a cost and performance advantage in this market.

The field of surveillance applications is a growing and important market nowadays. Live monitoring is one application in the surveillance scenario. Different control situations have to be supported, e.g., a surveillance room with a dedicated (or corporate) high data-rate network connection, a guard on the move connected via a limited data-rate network (e.g., WLAN) using a PDA with small display or a remote location with limited data-rate (e.g., DSL or a 2.5/3G network). Each control situation has specific requirements and constraints. Again, a single scalable encoding in combination with adaptation serves all needs, avoiding the effort of multiple encoder runs for the same view. Furthermore, the surveillance content can be stored more efficiently. If a reduction of the storage size is required and a degradation of the quality is acceptable, then a predefined basic

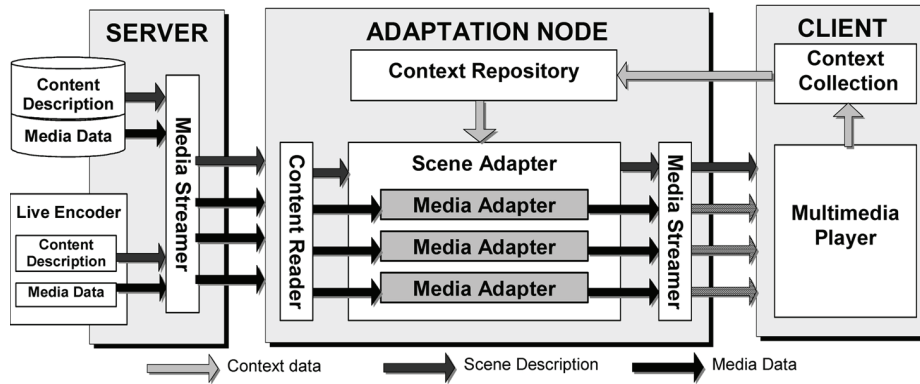


Fig. 1. Example for the adaptation architecture.

quality can be retained by removing only some enhancement layers of the video stream.

This paper presents an adaptive real-time SVC system based on MPEG-21 including encoding, error protection, adaptation, and decoding, which was implemented in the DANAE project [5].

The paper is organized as follows. In Section II, the applied architecture for adaptive video streaming with SVC is presented. Section III outlines the adaptation task based on the MPEG-21 multimedia platform including a real-time implementation of SVC. An unequal erasure protection (UXP) scheme for SVC transmission over error prone channels is presented in Section IV. In Section V simulation results for the real-time implementation as well as for the UXP scheme are presented. The paper is concluded in Section VI.

II. DANAE SVC STREAMING ARCHITECTURE FOR NETWORK AND TERMINAL ADAPTATION

A. DANAE Platform

To demonstrate the feasibility of adaptation of scalable multimedia content a platform for MPEG-21 multimedia content adaptation has been designed and implemented. The platform allows for adaptation of real-time encoded video streams as well as adapted representation of interactive scenes mixing text, 2-D and 3-D avatars, audio and video including session mobility between terminals with different capabilities. A simplified view of the DANAE architecture platform is depicted in Fig. 1. In the following, a brief description of the most relevant components for the SVC streaming application is provided. Among not further described parts, we can list MPEG-21 Digital Item Declaration (DID) processing, service tracking, or the MPEG-21 Digital Rights Management (DRM) license server.

There are three major components in the architecture: a client, a server and, in between, an adaptation node.

- The client collects user context and requests multimedia content to the adaptation node.
- The adaptation node relays the media requests to the server (or another adaptation node), receives and adapts the media units and finally sends them to the client.
- The server fulfills requests from the adaptation node. It reads media packets and metadata from either a storage

area or from a live encoder that produces scalable media packets and associated metadata on the fly.

The multimedia content is adapted according to a user context that contains the network characteristics, the user preferences and the terminal characteristics. The adaptation involves a client that collects the user context and embeds a multimedia player that requests multimedia content. The adaptation decisions and actions are taken by an adaptation node or a server that is able to modify the way content is presented. This can be done e.g., by changing the bit rates, the media type, or the layout of a scene. For instance, a rich media scene adapted for a PC may have a horizontal layout including large portions of text and high bit rate video. The same scene adapted for a PDA may have a vertical layout with audio instead of text and low bit rate video. Bandwidth consuming content like video may be adapted according to the actual bandwidth available.

As an initial context is needed for adapting the scene, the client collects all user and terminal profile information available. The information of this context can be static like terminal capability, network class and user preferences, or it can be dynamic information like the network bandwidth currently available. Once collected, the context is sent to the adaptation node and stored in a repository that keeps the data available for content adaptation.

When the player asks the adaptation node for a specific multimedia content, an MPEG-21 based representation of this content is used to compute the best adaptation according to the data stored in the context repository.

The adaptation benefits from the MPEG-21 description of the media with associated metadata information. For example directives how to proceed for scaling down specific media packets and what is the resulting quality (e.g., a frame of a video with multiple spatial layers). Another benefit of MPEG-21 is the possibility to include processing of a content description inside the description itself, allowing for a first level of specific adaptation [11].

The scene adapter is the key element of the adaptation architecture. First, it has to select the right media representation according to the user preferences (like audio versus text), terminal capability (like available codecs or video size matching the devices) and available network bandwidth and decides on the bandwidth allocated to each media stream processed. Finally, the scene adapter generates a presentation scene with an

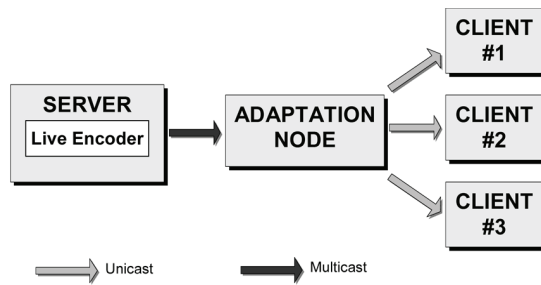


Fig. 2. Architecture for live encoding and transmission of SVC video.

adapted layout that refers to the selected media. Once the scene adaptation process is done, the scene adapter creates and initializes a dedicated media adapter for each media stream (e.g., audio, video) being adapted. Once everything has been properly configured and initialized, the media delivery and adaptation can start from the server to the multimedia player via the adaptation node.

During the session, the context may vary over time as for example the network characteristics may change or the player performance may decrease. This will induce a context modification at the client, which in turn will forward this information to the adaptation node via the context repository. The provided information can be used e.g., to re-allocate a new bandwidth to each media by modifying the relevant media adapter settings. Furthermore, if the changes are severe, the scene adapter may compute a new configuration including eventually new or alternative media formats. For instance, a video can be replaced by an images slideshow or an audio track can be replaced by text.

Besides the presented configuration, the architecture also supports a simpler scenario, where the scene consists only in an audio and a video stream, coming from a live source. The role of the adaptation node is simplified and mostly consists in relaying, with adaptation, a multicast live session to several unicast sessions. However, the media of each unicast session are adapted according to each bandwidth and terminal context. Fig. 2 shows the usage of a live encoder integrated with the server. The audio-visual stream is multicasted to the adaptation node that relays it with specific adaptation parameters (e.g., lower quality or frame rate) to multiple clients through unicast connections.

The DANA platform is thus generic enough to provide the appropriate test bed for different kind of investigations, and in particular concerning scalable video coding.

B. DANA Real-Time SVC Encoder

DANA partners contributed to the development and standardization of the scalable extension of H.264/AVC as a high-efficient, fully scalable video codec [1], [2], [4]. SVC allows for spatial, temporal, and quality [signal-to-noise ratio (SNR)] scaling. Scaling operations along these three dimensions of the “scalability cube” can be combined according to the scenario at hand. The SVC stream is organized in Network Abstraction Layer (NAL) units that convey the layer information. The SVC NAL unit header contains the spatial, temporal, and quality coordinates of the NAL unit payload in the scalability cube, which is used for identification and scaling operations of the NAL units. Note that the work described here is based on the SVC status as of mid-2006, where fine grain scalability (FGS) was

still included in the SVC specification. In the final SVC specification, FGS is replaced by medium grain scalability (MGS), see [2].

Essentially, the coding structure is a multiresolution pyramid, where each spatial layer is encoded using an individual core encoder based on H.264/AVC, see Fig. 3. As an enhancement of H.264/AVC, layers of lower spatial resolutions predict layers of higher resolutions. Temporal scaling is achieved by using the concept of hierarchical B-frames, [2]. For quality scaling, two modes are provided by SVC: A coarse grain scalability (CGS) mode including inter-layer prediction and an FGS mode providing progressive refinement of the prediction residual. For changing the spatial resolution, CGS layers are employed with additional interpolation for the motion or texture prediction as applicable. Up to three FGS layers, usually representing a refinement by a factor of two each, can be assigned to each CGS or spatial layer. Quality scaled versions of arbitrary bit rates can be extracted by applying simple truncation operations to the FGS layers. Rate-distortion optimized truncation can be achieved through the application of quality layers [6].

According to some scenarios described in Section I, the SVC encoder in a complete end-to-end chain should be real-time capable. Since the JSVM reference encoder [20] is far from being real-time capable, thorough investigations have been undertaken in order to improve the run-time performance. JSVM version 3.3.1 served as the starting point for the real-time developments. A hot-spot analysis of the reference encoder resulted in a list of encoder modules, which contribute most to the overall computational effort. Starting from this set of computational hot-spots, a selection of time consuming functions have been replaced by assembly code involving processor specific command sets. Mainly, the following modules have been optimized.

- Motion vector search: Essentially the sum of absolute differences (SAD) calculation has been accelerated.
- Quarter-pel filter: This filter calculates interpolations in order to achieve motion compensations with quarter-pel accuracy.
- The 2-D spatial up-sampling filter used to generate interpolations from lower spatial levels to higher levels for inter-layer intra-predictions, see Fig. 3.

Essentially, these modules have block-processing structures, which are amenable for code optimization involving SIMD commands. Apart from potential rounding errors, these code optimizations do not decrease the coding efficiency, since no algorithmic changes have been made.

As the gain in run-time performance due to these modifications is not sufficient, additional efforts were made to simplify the encoding algorithm. To this end, the motion estimation part of the encoder has been further investigated. It is well-known that the H.264/AVC standard offers a high degree of flexibility in the selection of the block sizes and shapes for the motion compensation [7]. A multitude of different prediction schemes are available and the selection is made by minimizing a certain cost criterion. Since this optimization process is computationally very demanding, the relation between the saved computational effort versus the decrease in coding efficiency was investigated. This effort led to a scheme where predictions are calculated according to a certain predefined order of prediction modes depending on the predicted computational effort. When

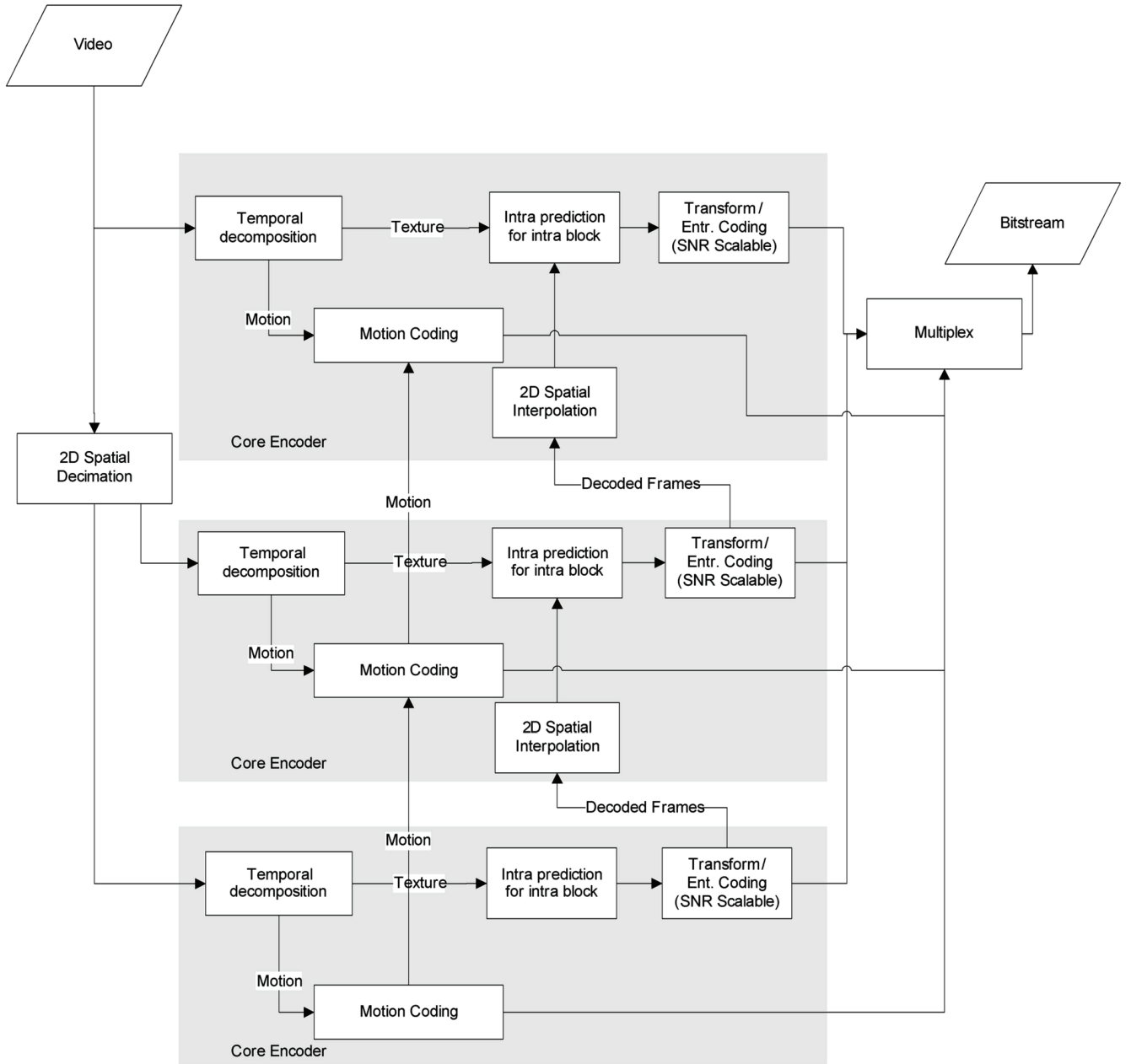


Fig. 3. SVC encoder using a multiresolution pyramid with three levels of spatial scalability.

the required processing power is predicted to be too high then the optimization process is stopped and the mode corresponding to the best result so far is taken. As a consequence, the set of employed prediction modes is reduced when the computational burden is high. These algorithmic modifications assure a standard compliant bit stream, however, the computational complexity has been significantly reduced and the real-time constraint has been fulfilled.

III. SVC STREAM ADAPTATION

A. Adaptation: The Objective and Its Constraints

The main objective for SVC stream adaptation is to ensure optimum video quality for a given set of constraints where these

constraints may be either static after the session set up or may dynamically change over the session duration.

It can be seen from the various application scenarios described in Section I that typical constraints imposed during the usage comprise:

- terminal capabilities like screen size or processing power (i.e., display resolution and supported profile/level combination);
- network capabilities like the maximum bandwidth and, as an example for a dynamically changing constraint, network status information like the currently available bandwidth or the packet loss ratio;
- optionally also user related constraints like a personal preference indication for temporal resolution versus spatial detail.

A very different second type of constraints is imposed by the SVC encoding process. Here, in a tradeoff between flexibility and compression efficiency, it is decided, which adaptation options shall be applicable to the encoded stream, e.g.:

- extractable spatial and temporal resolutions;
- achievable bit rates using CGS or FGS for SNR scalability;
- optionally also the resulting quality measured.

For content with highly varying motion intensity or detail intensity, these bit-stream-related constraints may vary over time. Changes to this information set will also occur in services like (mobile) TV broadcast whenever a program changes.

For deciding the actual adaptation to be performed, the constraints from the former group need to be matched to the constraints from the second group. In case of possibly dynamically changing constraints, this matching process must be repeated during the session. For each update, the process will result in clear decisions for the adaptation process itself, i.e., which NAL units should be sent or dropped.

In order to build end-to-end services, both, the constraints information as well as the adaptation information, need to be described in an interoperable format. In the SVC syntax itself, there are syntax elements in the NAL unit header, in the sequence and picture parameter sets and additional supplemental enhancement information (SEI) messages, which can carry the constraint information related to the bit stream. The scalability coordinates (the SVC syntax elements *dependency_id*, *temporal_level* and *quality_level*) and the priority information (*priority_id*) in the NAL unit header provide the information which NAL units should be dropped according to an adaptation decision. For the usage related constraints, other description formats beyond the SVC syntax are needed, e.g., like the User Agent Profile (UAProf) used in mobile telephony services [8] or the DIA UED described in the following.

However, directly using the in-band information in the SVC stream has further implications. 1) Any adaptation decision unit and any adaptation engine needs to understand the SVC syntax and needs to parse the SVC stream. 2) For services with multimedia content, different mechanisms have to be deployed to adapt at least the audio and the video streams. In addition, the decision taking may need to take into account a tradeoff between audio and video quality and hence additional constraint information on session level has to be provided. Therefore, there are good reasons to further explore description formats that support codec and media independent adaptation mechanisms. To our knowledge, the only complete specification satisfying these requirements is the MPEG-21 standards suite.

B. MPEG-21 DIA: Tools for Media Adaptation

The overall aim of the MPEG-21 standard [9]—the so-called Multimedia Framework—is to enable transparent and augmented use of multimedia resources across a wide range of networks, devices, user preferences, and communities. For the media adaptation aspects and for this paper, we concentrate on MPEG-21 Part 7 (DIA) [10], [13]. DIA specifies XML based description tools to assist with the adaptation of multimedia content. This means that tools used to control the adaptation process are specified, but the exact implementation of an adaptation engine is left open to industry competition.

The relevant DIA description tools for the constraints are as follows.

- For the usage related constraints, the Usage Environment Description (UED): UEDs are a large collection of relatively simple XML constructs for capturing all kinds of environment descriptions including those relevant for adaptation decisions. The covered properties range from user characteristics (e.g., preferences, impairment, usage history) to terminal capabilities, network characteristics and the natural environment (e.g., location, time, illumination).
- For the bit stream related constraints, the Adaptation Quality of Service (AQoS) and the Universal Constraint Descriptor (UCD): AQoS describes for a given bit stream the relationship between the adaptation parameters and constraints, resulting resource characteristics, quality, and possibly other parameters. In addition, the UCD can also be used to declare an optimization function to control the selection of the best adaptation option. Further information on the functionality of these descriptors in the decision taking process can be found in [10]. To cope with varying bit stream characteristics, the AQoS can be fragmented into so-called *Adaptation Units (ADUs)*. An ADU describes the adaptation options and resulting characteristics for a certain part of the bit stream.
- For the control of the actual adaptation process, the generic Bit Stream Syntax Description (gBSD), and the BSD Transformation description: The gBSD represents an abstract and high level XML description of a bit stream syntax, mainly providing information about the bit stream structure. It also includes references to and into the described bit stream. Each described syntax element is represented by a gBSD Unit that provides a handle, which can be annotated and then be linked to the output parameters of the AQoS. Modifications to this XML based gBSD can be directly mapped to modifications to the bit stream. They are controlled by the BSD Transformation. The standard does not explicitly fix a particular transformation, but the practical default is XSLT, the most common standardized XML transformation [13]. An XSLT sheet steers the transformation process by using the output parameters of the AQoS as input, matching them to the annotations in the gBSD, and performing the described modifications.
- For the coupling of all descriptions, the BSDLink: The BSDLink provides references to the AQoS, to the gBSD, and to the XSLT that correspond to a single adaptation description. It should be noted that the AQoS and the gBSD have to be produced for each SVC stream, preferably during the encoding of the SVC. The XSLT sheet is stream independent and can be generated once for a given usage scenario.

Based on these descriptions, the whole adaptation process is abstracted from the bit stream syntax and format specifics, i.e., a generic adaptation engine can be built, which is suitable for any media stream.

A block diagram for a generic adaptation engine as implemented in the DANA platform is shown in Fig. 4 depicting the decision taking and adaptation process (see also [14]). The adaptation decision taking engine (ADTE) selects the best transfor-

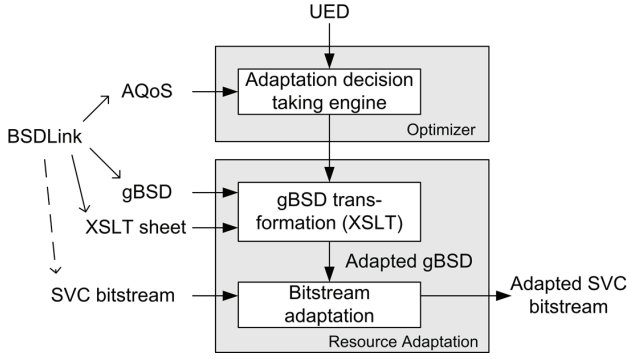


Fig. 4. Adaptation engine.

```

<dia:DIA xmlns:dia="urn:mpeg:mpeg21:2003:01-DIA-NS"
[...]>
  <dia:Description
xsi:type="dia:UsageEnvironmentType">
    [...]
    <dia:DisplayCapability
xsi:type="dia:DisplayCapabilityType">
      <dia:Mode>
        <dia:Resolution horizontal="176"
vertical="144" /

```

Fig. 5. UED example.

mation from the possibilities in the AQoS under the constraints found in the UED. Next, the original gBSD is transformed according to the adaptation decisions in the optimizer resulting in an adapted gBSD description. This describes the relevant parts in the SVC bit stream, which are accordingly adapted by the bit stream adaptation module. The latter two process steps can be merged in an optimized implementation.

C. SVC Adaptation Based on MPEG-21 DIA Tools

The adaptation process exploits inherently supported features of scalable media formats. Taking advantage of the scalability features of SVC for the use cases addressed in the DANA platform allows a significant simplification of the generic tools and processes specified in MPEG-21 DIA.

From the UED, the relevant subset is reduced to the display capability, codec capability, network capability, network condition, and user characteristics. An extract of an example instance for the display capabilities is given in Fig. 5, where a display size of 176×144 pixels (QCIF format) is defined. (Note that the most interesting parts in the XML examples of Figs. 5–7 are printed in bold.)

In a very simple case, the AQoS descriptor would directly match the available bandwidth on the transmission channel with the target bit rate of the video. A more sophisticated AQoS may describe the impact of temporal and spatial scalability on the target bit rate as shown in the (shortened) example in Fig. 6. Here, the described SVC stream provides one spatial layer (“S”), four temporal layers (“T”) and five FGS truncation points per FGS/MGS layer (“F1,” “F2”), resulting in 11 bit rates per spatio-temporal resolution (including the CGS layer)

```

<DIA xmlns="urn:mpeg:mpeg21:2003:01-DIA-NS" [...]>
[...>
  <Description xsi:type="AdaptationQoSType">
    <Module xsi:type="LookUpTableType">
      <Axis iOPinRef="S">
        <AxisValues xsi:type="IntegerVectorType">
          <Vector>1</Vector>
        </AxisValues>
      </Axis>
      <Axis iOPinRef="T">
        <AxisValues xsi:type="IntegerVectorType">
          <Vector>0 1 2 3</Vector>
        </AxisValues>
      </Axis>
      <AxisRef iOPinRef="F1"/>
      [...]
      <Content iOPinRef="targetBitrate">
        <ContentValues xsi:type="IntegerMatrixType"
mpeg7:dim="1 4 11">
          <Matrix>13 26 31 36 41 46 53 61 69 77 85
29 35 42 49 55 [...] 176 205 234 262 </Matrix>
        </ContentValues>
      </Content>
    </Module>
    <Module xsi:type="LookUpTableType">
      <AxisRef iOPinRef="F1"/>
      <Content iOPinRef="F2">
        <ContentValues xsi:type="IntegerMatrixType"
mpeg7:dim="11">
          <Matrix>0 0 0 0 0 0 20 40 60 80
100</Matrix>
        </ContentValues>
      </Content>
    </Module>
    <IOPin id="PSNR"/>
    <IOPin id="targetBitrate" semantics="urn:mpeg:
mpeg21:dia:cs:MediaInformationCS:9"/>
    <IOPin id="S"/>
    <IOPin id="T"/>
    <IOPin id="F1">
      <Axis>
        <AxisValues xsi:type="IntegerVectorType">
          <Vector>0 20 40 60 80 100 100 100 100 100
100</Vector>
        </AxisValues>
      </Axis>
    </IOPin>
    <IOPin id="F2"/>
  </Description>
</DIA>

```

Fig. 6. AQoS example.

and 44 bit rates in total (see “targetBitrate” in Fig. 6). The resulting bit rates are listed in a three-dimensional matrix. Note that the matrix entries are mapped into a one-dimensional array, starting with incrementing the “F2” and “F1” index first, then index “T.”

In the case of SVC streams, the gBSD will preferably describe and reference entire NAL units. In addition to the start point and the length of the NAL unit, the indexes for the scalability axes are defined as depicted in Fig. 7. Besides the temporal level (“T”) and spatial level (“S”), also the FGS/MGS layer (“F”) and the NAL unit type (“N”) are described in the gBSD. For the FGS layer, “F0” indicates the CGS layer, “F1” and “F2” the first and the second FGS layer, respectively. For the NAL unit type, “N1” indicates non-IDR AVC NAL units, “N20” non-IDR SVC NAL units and “N6” SEI messages.

The XSLT process is reduced to very simple pattern matching and copying in the gBSD. The modifications to the gBSD are then directly reflecting the required modifications to the bit

```

<dia:DIA [...] xmlns="urn:mpeg:mpeg21:2003:01-DIA-
gBSD-NS" [...]>
  <dia:Description xsi:type="gBSDType"
addressUnit="bit" addressMode="Absolute"
bs1:bitstreamURI="cdi_qcif125_soccerx10_201.raw">
    [...]
    <gBSDUnit length="55872" start="28160"
marker="access_unit">
      <gBSDUnit start="28160" length="72"
marker="T0:S0:F0:N6"/>
      <gBSDUnit start="28232" length="5776"
marker="T0:S0:F0:N1"/>
      <gBSDUnit start="34008" length="8216"
marker="T0:S0:F1:N20"/>
      <gBSDUnit start="42224" length="13064"
marker="T0:S0:F2:N20"/>
      <gBSDUnit start="55288" length="72"
marker="T1:S0:F0:N6"/>
      <gBSDUnit start="55360" length="1264"
marker="T1:S0:F0:N1"/>
      <gBSDUnit start="56624" length="1744"
marker="T1:S0:F1:N20"/>
      <gBSDUnit start="58368" length="5648"
marker="T1:S0:F2:N20"/>
    [...]
  </gBSDUnit>
  [...]
</dia:Description>
</dia:DIA>

```

Fig. 7. gBSD example.

stream that in turn are executed by the adaptation engine. As stated before, this can be performed by simply dropping NAL units for the adaptation of SVC streams.

Real-time evaluations of the adaptation process in the DANAE streaming platform have shown that, even without having put much emphasis on software optimization, more than 20 different SVC streams can be processed in parallel by the adaptation and streaming engines on a standard laptop (Pentium Centrino, 1.6 GHz). For the adaptation process, the number of gBSD units to be processed, i.e., the number of NAL units, was identified to be the determining performance criterion. In other words, the number of enhancement layers in the full (i.e., not adapted) SVC stream is more relevant than the bit rate. Another observation was that the largest portion of the processing time in the adaptation engine is consumed by the XSLT. Currently, a generic XSLT processor is used for the gBSD transformation. It can be expected that specialized transformation processes would lead to significant performance improvements. In any case, compared to transcoding of non-scalable content, adaptation dramatically saves computational resources in the server and in the network (e.g., on media gateways).

IV. UNEQUAL ERASURE PROTECTION FOR SVC

For improved robustness of SVC transmission over error prone channels using RTP, a UXP scheme for SVC over RTP has been developed. The scheme applies a specific payload format, since the introduced parity information and interleaving prevent the application or extension of the existing RTP payload format for H.264/AVC, or the proposed payload format for SVC [17], [18]. The scheme presented here is based on a proposal to the IETF, which employs Reed–Solomon codes for the generation of parity information [15]. The general concept of the approach was originally presented in [16]. The scheme allows for the localization of losses and employs the erasure

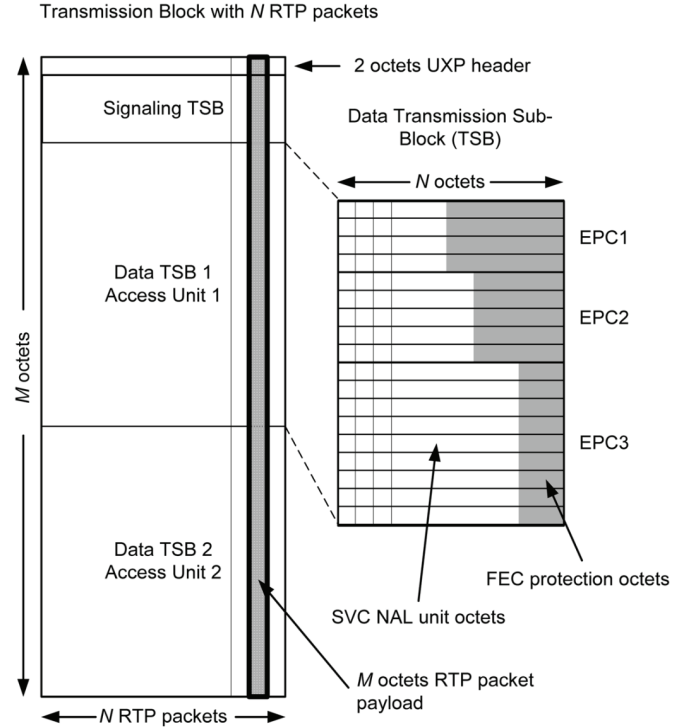


Fig. 8. Example of a transmission block containing a signaling transmission subblock with UXP information on the protection classes and two data transmission subblocks with two Access Units. In the data transmission subblocks, each NAL unit is assigned an amount of parity information (FEC protection) according to the EPC it belongs to.

correcting features of the employed Reed–Solomon codes for reconstruction of the erased information.

1) *Outline of the UXP Concept:* The basic approach of UXP is to generate an interleaved protected media stream, where for each layer of the original media stream an adjustable amount of forward error correction (FEC) or parity information is added. The media stream is organized in transmission blocks, where the information is interleaved and distributed over a configurable number of packets. As the interleaving is concentrated on a small number of access units, the transmission delay added by the scheme can be controlled. If each transmission block is bound to convey exactly one SVC Access Unit, the transmission delay can be equivalent to the transmission delay of an un-protected stream. In Fig. 8, a schematic presentation of the structure of a transmission block and the applied concept of interleaving is presented. For protection the bytes of the NAL units and the protection information are written “horizontally” to transmission block while for transmission, the transmission block is read vertically to provide interleaving (see Fig. 8).

The number of RTP packets that belong to one transmission block is configurable. All RTP packets that belong to one transmission block have the same payload size which is configurable as well, but bound by the maximum packet size (MTU size) of the transport channel. A transmission block consists of one signaling transmission subblock and one or more data transmission subblocks. Each data transmission subblock is assigned an Access Unit, where each NAL unit within the Access Unit is assigned a configurable erasure protection class (EPC) as depicted in Fig. 8. The ability to include multiple Access Units

in one transmission block can improve the exploitation of the available packet size, especially in case of videos encoded at very low bit rate.

The rows of each erasure protection class in a data transmission subblock are filled with the octets of a NAL unit, and for each row the corresponding number of parity octets is included. If the NAL unit octets do not fill the last row of an EPC and the EPC of the following NAL unit provides equal or less protection, the remaining space can be filled with octets from the following NAL unit. Otherwise, stuffing octets are introduced.

The signaling transmission subblock is generated after the data transmission subblock(s) have been established. It conveys information on the redundancy profile, i.e., the size and the protection information of the EPCs, applied to the data transmission subblocks. Additionally, the presence of stuffing octets is indicated. This transmission subblock receives the strongest protection as it contains the most sensitive information for the whole transmission block. For a detailed description, the reader is referred to [15].

In the presented UXP scheme for SVC, syntax elements of the SVC NAL unit header including NAL unit type, priority_id, dependency_id, temporal_id, and quality_id are considered for erasure protection class identification. The applicable configuration of the error protection class strongly depends on the actual transmission conditions, which may vary over time. A possible method for deriving optimized protection configuration is presented, e.g., in [19].

For transmission in RTP packets, the protected data of a transmission block is interleaved as depicted in Fig. 8. An additional two-octet UXP header is inserted at the beginning of the RTP packet payload that enables identification of the UXP packets assigned to each transmission block. It contains the payload type of the protected media stream (here SVC) and a transmission block indicator which depends on the RTP sequence number. Either the least significant octet of the RTP sequence number of the first RTP packet of the current transmission block, or the total number of RTP packets for the current transmission block is indicated. Based on the RTP sequence number of the current packet and the transmission block indicator in each UXP header, the receiving entity is able to recognize both transmission block boundaries and the actual position of packets (both received and lost ones) in the transmission block.

As described in Section III the generic bit stream description (gBSD) associated with each SVC stream describes exactly the structure of the scalable stream in terms of NAL units. After an adaptation operation where certain layers of the scalable bit stream have been discarded, the transformed gBSD describes the structure of the adapted stream. Therefore, using this structural information, the error protection code can be generated for each remaining layer, for which protection is desired. Similar to the stream adaptation itself, the generation of the error protection code can be performed statically, i.e., once for the whole stream, or dynamically, e.g., access unit by access unit. Moreover, in distributed adaptation scenarios, i.e., where several successive stream adaptations take place along the end-to-end chain, the gBSD is able to carry the structural information of both the SVC bit stream data as well as the associated error protection data.

2) *Reconstruction of Error Prone RTP Streams:* On the receiver side, the incoming RTP packet stream is buffered until the start sequence number and the size of a transmission block can be determined. After inserting the payload of all related RTP packets into the transmission block, the signaling transmission subblock containing the description of the erasure protection classes is recovered. If the signaling transmission subblock cannot be recovered due to a loss rate higher than the applied protection, the whole transmission block has to be discarded. If the packet loss rate exceeds the protection of single erasure protection classes, the corresponding NAL units cannot be recovered and have to be discarded. A careful design of the applicable redundancy profile is required to prevent the loss of essential NAL units (e.g., in the base layer) before nonrequired enhancement layer information is lost.

V. RESULTS

A. SVC Real-Time Rate-Distortion Results

The optimized encoder is capable of encoding sequences at frame rates of 25 fps (QCIF) and 12.5 fps (CIF) on a high-end PC with Intel Pentium D processor.

In order to investigate the coding efficiency of the real-time encoder, the rate-distortion performance has been compared to the standard reference software. Five well-known test-sequences have been encoded/decoded using the JSVM 3.3.1 reference codec and compared to the respective results obtained by using the optimized encoder. In all simulations, CABAC was used. Two FGS layers (F1, F2) have been encoded for CIF and QCIF resolutions and four dyadic temporal levels. The temporal levels result in frame rates of 15, 7.5, 3.75, and 1.875 fps. The two FGS layers have been truncated in ten equidistant steps (five for each) yielding a total of 11 rate points (including the base layer) for each temporal level. Thus, for each spatial resolution, there are 44 extraction points in the temporal/quality scaling space. A short GOP-length of eight frames and a short intra-frame interval of one GOP-length were chosen in order to limit the overall delay and to provide fast random access, which are important requirements for IPTV and video phone/conferencing applications.

Figs. 9 and 10 show the calculated PSNR values of the 44 extraction points for the Mobile sequence. As can be seen, the differences in PSNR are very low for the lower temporal resolutions and increase (though moderately) for the higher temporal resolutions.

These observations are similar for the other test-sequences that have been compared although the sequences are very different in terms of motion characteristics, contrast and brightness variations. This can be seen from Table I, where the results of all test-sequences are listed. In order to compare the rate-distortion curves of the two encoders, a slightly modified version of the Bjontegaard average peak SNR (PSNR) difference was used [21]. This measure calculates an approximation of the average difference in the PSNR curves versus rate

$$\overline{\Delta\text{PSNR}} = \frac{1}{b-a} \int_a^b (p_2(r) - p_1(r)) dr, \quad p_2(r) \geq p_1(r) \text{ all } r$$

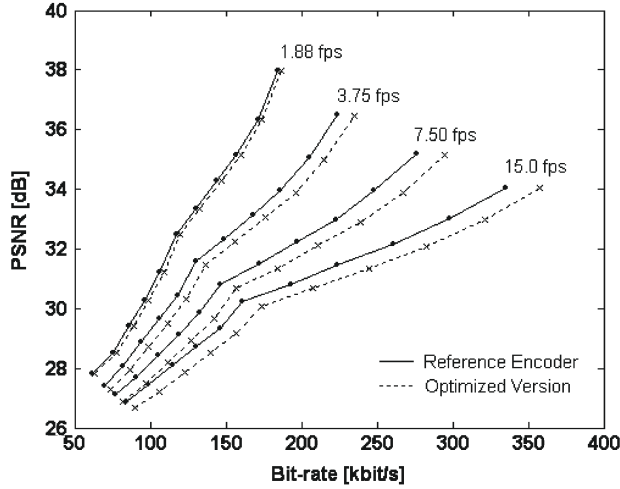


Fig. 9. Rate-distortion curves of the Mobile sequence (QCIF resolution) for 4 temporal layers each consisting of 11 rate points.

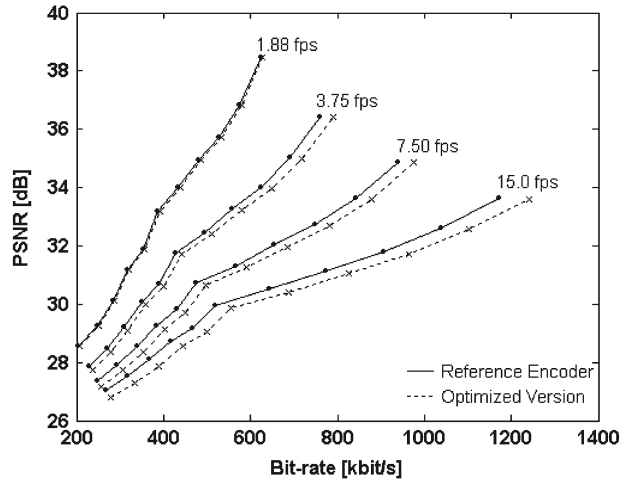


Fig. 10. Rate-distortion curves of the Mobile sequence (CIF resolution) for four temporal layers each consisting of 11 rate points.

where $r = \log(\text{rate})$, p_2 , and p_1 are cubic polynomials approximating the two PSNR curves in a least squares sense and the interval $[a, b]$ defines the intersection of the log-rate ranges of the two curves.

As can be seen from Table I, the maximum Average PSNR Differences are quite low, i.e., 0.58 dB for City (QCIF) and Mobile (QCIF) and 0.4 dB for Mobile (CIF).

B. UXP Performance

The performance of the presented UXP scheme is demonstrated for a SNR scalable SVC stream that comprises an H.264/AVC base layer and two FGS layers. The GOP size of the stream is 32. CABAC is used for entropy coding. The SVC reference software JSVM 6.7 was employed for the test. Four erasure protection classes are defined.

- EPC0: Sequence and Picture Parameter Sets and Scalability SEI messages (always 60% parity).
- EPC1: quality_id equal to 0, i.e., the quality base layer.
- EPC2: quality_id equal to 1, i.e., the first FGS layer.
- EPC3: quality_id equal to 2, i.e., the second FGS layer.

TABLE I
AVERAGE PSNR DIFFERENCES (dB)

Sequence		Temporal Level			
		1.88 fps	3.75 fps	7.5 fps	15.0 fps
Crew	QCIF	0.0	0.1	0.35	0.37
	CIF	0.0	0.11	0.13	0.21
City	QCIF	0.01	0.41	0.58	0.55
	CIF	0.0	0.23	0.25	0.26
Soccer	QCIF	0.12	0.09	0.07	0.34
	CIF	0.0	0.04	0.10	0.20
Mobile	QCIF	0.23	0.5	0.58	0.55
	CIF	0.07	0.29	0.34	0.40
Harbour	QCIF	0.01	0.25	0.30	0.29
	CIF	0.08	0.21	0.17	0.16

Average PSNR Differences between reference and optimized encoder for each temporal level and spatial resolution. The most critical sequence in this set is the Mobile sequence resulting in Average PSNR Differences of 0.58 dB (QCIF) and 0.4 dB (CIF).

The amount of parity information p_i for the erasure protection classes EPC i , $i = 1, 2, 3$ was arranged such that $p_1 > p_2 \geq p_3$, and a constant transmission rate was met. For demonstration of the performance of the UXP scheme, results are provided for an equal erasure protection (EEP) configuration, where a constant protection is applied to all video coding layer NAL units. The RTP stream was exposed to random packet loss with increasing loss rate and the PSNR for the reconstructed video sequence was measured. For each loss rate, the transmission and recovery experiment was repeated 50 times. The average PSNR resulting from these repeated tests is reported below.

In the scenario described here, no error concealment methods were implemented in the decoder. Therefore, the results for the erasure protection are not influenced by error concealment strategies at the decoder. Due to the absence of error concealment, the loss of Access Units, or NAL units of lower layers (e.g., base layer, intermediate quality layers), may lead to streams which cannot be decoded by the JSVM decoder. In the simulations, reconstructed streams that showed this issue were dropped and only reconstructed streams decodable by the JSVM were regarded for the PSNR measurements. Here, results are presented that provided a successfully decoded video stream for at least 80% of the 50 conducted test repetitions. Fig. 11 shows the PSNR over packet loss rate for the sequence BUS at QCIF and 15 fps, encoded with 2 FGS layers at a maximum bit rate of 360 kbps.

In the given example, the transmission rate was configured to correspond to a rate increase of approximately 33% for the applied protection for both, UXP and EEP. The presented protection configurations do all provide the same transmission rate, subject to a tolerance of $\pm 2\%$.

From Fig. 11, it can be seen that up to a loss rate of roughly 10% the presented scheme can provide configurations that perform within 0.5 dB compared to the quality measured without

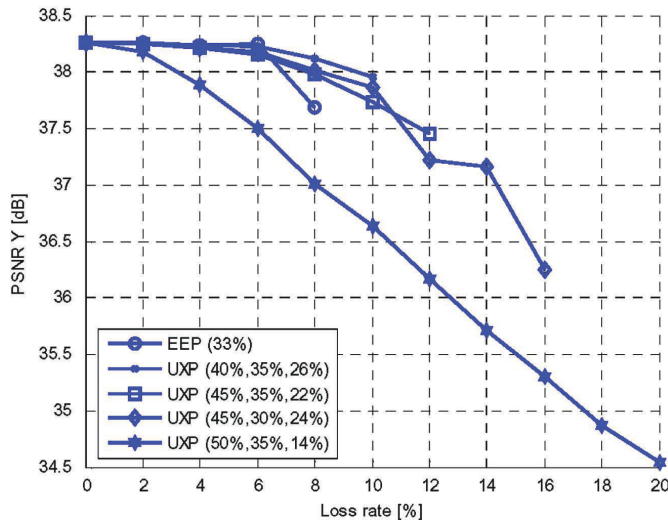


Fig. 11. PSNR over packet loss rate for the test sequence BUS QCIF 15 Hz, encoded with two FGS layers. UXP with three erasure protection classes (EPC1, EPC2, EPC3) for the three available quality layers is compared to EEP with a fixed protection rate for all quality layers.

losses. It can be seen that compared to the EEP configuration, the decodability of the stream with UXP protection is improved for cases with higher losses. Depending on the amount of anticipated transmission loss, configurations can be selected that provide graceful degradation of the reconstructed quality up to a loss rate of 20% in the given example.

The results reveal that the decodability of the stream strongly depends on the protection applied to the base layer. The higher the base layer protection, the higher the probability that lost essential packets can be recovered. Depending on the amount of protection for the enhancement layers, the distortion characteristics are controlled. The more protection is applied to the enhancement layers, the better the reconstructed quality under higher loss rates. The curves in Fig. 11 reveal that the amount of protection for the highest FGS layer has a strong impact on the resulting PSNR performance. This relates to the amount of SVC bit rate spent for this layer (approximately half of the total rate).

VI. CONCLUSION

We presented the integration of SVC into a generic platform for multimedia adaptation. With this platform SVC and other media can be adapted according to the MPEG-21 framework. The scheme comprises a real-time SVC implementation and includes provisions for protected SVC transmission in an error prone RTP environment. Simulation results demonstrated the remarkable performance of the presented SVC real-time encoder implementation and revealed the benefits and the adaptability of the proposed UXP scheme according to varying transmission loss rates.

ACKNOWLEDGMENT

The authors would like to thank in particular F. Sanahuja (Siemens Corporate Technology), O. Klüsener (T-Systems Enterprise Services), and I. Wolf (T-Systems Enterprise Services) for their work and discussion on the adaptation platform. All the DANAE project consortium partners, contributing to the development and implementation of the described system, are gratefully acknowledged. They further gratefully thank the anonymous reviewers for their very constructive comments and suggestions.

REFERENCES

- [1] J. Reichel, H. Schwarz, and M. Wien, *Joint Scalable Video Model JSVM-7: Joint Draft 7 with Proposed Changes*, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, Doc. JVT-T202, Jul. 2006.
- [2] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.
- [3] *Requirements and Applications for Scalable Video Coding v.5*, ISO/IEC JTC1/SC29/WG11, Requirements Group, Doc. N6505, Jul. 2004.
- [4] M. Wien, H. Schwarz, and T. Oelbaum, "Performance analysis of SVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1194–1203, Sep. 2007.
- [5] *DANAE Homepage*. (Mar. 2006) [Online]. Available: <http://danae.rd.francetelecom.com/>
- [6] I. Amonou, N. Cammas, S. Kervade, and S. Pateux, "Optimized rate-distortion extraction with quality layers," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1186–1193, Sep. 2007.
- [7] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.
- [8] *OMA User Agent Profile V2.0*. (Feb. 2006) [Online]. Available: http://www.openmobilealliance.org/release_program/uap_v2_0.html
- [9] I. Burnett, F. Pereira, R. Van de Walle, and R. Koenen, Eds., *The MPEG-21 Book*. New York: Wiley, 2006.
- [10] *Information Technology—Multimedia Framework (MPEG-21)—Part 7: Digital Item Adaptation*, ISO/IEC 21000-7, 2004.
- [11] *Information Technology—Multimedia Framework (MPEG-21)—Part 10: Digital Item Processing*, ISO/IEC 21000-7, 2006, 2004.
- [12] D. Mukherjee, E. Delfosse, J. G. Kim, and Y. Wang, "Optimal adaptation decision-taking for terminal and network quality of service," *IEEE Trans. Multimedia*, vol. 7, no. 2, pp. 454–462, Apr. 2005.
- [13] *W3C, XSL Transformations (XSLT) Version 1.0*. (Nov. 1999) [Online]. Available: <http://www.w3.org/TR/1999/REC-xslt-19991116>
- [14] A. Hutter, P. Amon, G. Panis, E. Delfosse, M. Ransburg, and H. Hellwagner, "Automatic adaptation of streaming multimedia content on a dynamic and distributed environment," in *Proc. ICIP*, Genova, Italy, Sep. 2005, pp. 716–719.
- [15] G. Liebl, M. Wagner, J. Pandel, and W. Weng, "An RTP payload format for erasure-resilient transmission of progressive multimedia streams," [Online]. Available: <http://tools.ietf.org/wg/avt/draft-ietf-avt-uxp/draft-ietf-avt-uxp-07.txt>, Oct. 2004
- [16] A. Albanese, J. Bloemer, J. Edmonds, and M. Luby, "Priority encoding transmission," *IEEE Trans. Inf. Theory*, vol. 42, no. 6, pp. 1737–1744, Nov. 1996.
- [17] S. Wenger, M. M. Hannuksela, M. Westerlund, and D. Singer, RTP Payload Format for H.264 Video, IETF RFC 3984, Feb. 2005.
- [18] S. Wenger, Y.-K. Wang, and T. Schierl, "RTP Payload format for SVC video," [Online]. Available: <http://tools.ietf.org/wg/avt/draft-wenger-avt-rtp-svc-02.txt> Jun. 2006
- [19] U. Horn, K. Stuhlmüller, M. Link, and B. Girod, "Robust Internet video transmission based on scalable coding and unequal error protection," *Image Commun.*, vol. 15, no. 1–2, pp. 77–94, Sep. 1999.
- [20] J. Vieron, M. Wien, and H. Schwarz, *JSVM-7 Software*, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, Doc. JVT-T203, Jul. 2006.
- [21] G. Bjontegaard, *Calculation of Average PSNR Differences Between RD-Curves* ITU-T SG16/Q6, VCEG-M33, Apr. 2001.



Mathias Wien (S'98–M'03) received the diploma and Dr.-Ing. degrees from RWTH Aachen University, Aachen, Germany, in 1997 and 2004, respectively.

From 1997 to 2004, he worked towards the Ph.D. as a Researcher at the Institute of Communications Engineering at RWTH Aachen University, where he is now employed as Senior Research Scientist and Head of Administration. His research interests are in the area of image and video processing, space-frequency adaptive and scalable video compression, and robust video transmission. He was an active contributor to the first version of H.264/AVC. He is a Co-Editor of the SVC amendment to H.264/AVC. He is an active contributor to the ITU-T VCEG and the Joint Video Team of VCEG and ISO/IEC MPEG where he co-chaired several AdHoc Groups.



Renaud Cazoulat received the diploma and Dr.-Ing. degrees in computer science from the University of Caen, Caen, France, in 1992 and 1996, respectively.

From 1992 to 1996, he worked mainly on artificial intelligence and the influence of stochastic systems on neural networks teaching. He joined France Telecom in 1997 to work on multimedia standard definition with a focus on the system part of MPEG-4. He co-founded Envivio in 2000, a company dedicated to provide end to end MPEG-4 video solutions. He came back to France Telecom in 2004 to lead a work package in the EU-IST project DANAE and to work on rich media systems for mobile environments and new networks like DVB-H and HSDPA.



Andreas Graffunder received the diploma and Dr.-Ing. degrees in electrical engineering from the Technical University of Berlin, Berlin, Germany, in 1986 and 1994, respectively.

From 1986 to 1994, he had a position as a Research Assistant with the Institute for Control Theory and Systems Dynamics of TU-Berlin, where he mainly worked in the fields of nonlinear robot control, vehicle dynamics and stereo vision. Between 1993 and 2002, he was with several companies and institutions where he led research and development projects in the fields of clinical evoked potential analysis, 3-D graphics, image segmentation, video coding, and video conferencing. In 2002, he joined T-Systems Enterprise Services, Berlin, Germany, where he currently has a position as a Senior Project Manager in the Department of Media Broadband and Entertainment Applications. He published scientific papers in several fields including robotics, computer vision, stochastic signal processing, image processing and multimedia communication. He contributed to the MPEG-4 standard since 1996. He has also been involved in several European research projects partly as a work package leader such as in the EU-IST project DANAE.



Andreas Hutter received the diploma and Dr.-Ing. degrees in communications engineering from the Munich University of Technology, Munich, Germany, in 1993 and 1999, respectively.

From 1993 to 1999, he was as a Research Assistant with the Institute for Integrated Circuits of the Munich University of Technology, where he mainly worked on algorithms for video coding and on the implementation of multimedia systems for mobile terminals. He joined Siemens Corporate Technology, Munich, Germany, in 1999, where he is currently leading the competence centre for video and multimedia communications. He has been an active member of MPEG since 1995 where he contributed to the MPEG-4, the MPEG-7, and the MPEG-21 standards. He was co-editor of the MPEG-7 Systems standard and he is acting as HoD (Head of Delegation) of the German National Body at MPEG. He has also been actively involved in several European research projects, where he has been work package leader of the EU-IST projects ISIS and DANAE.



Peter Amon received his Dipl.-Ing. (M.Sc.) degree in electrical engineering from the University of Erlangen-Nuremberg, Germany in 2001, where he specialized in communications and signal processing.

In 2001, he joined Siemens Corporate Technology, Munich, Germany, where he is currently working as a Research Scientist in the Networks and Multimedia Communications Department. In this position, he is and has been responsible for several research projects. His research field encompasses video coding, video transmission, error resilience, and joint source channel coding. In that area, he has authored or co-authored several conference and journal papers. He is also actively contributing to and participating at the standardization bodies ITU-T and ISO/IEC MPEG, where he is currently working on scalable video coding and the respective storage format.