Visual Comput (2008) 24: 735–743
DOI 10.1007/s00371-008-0255-7

ORIGINAL ARTICLE

Saar De Zutter
Mark Asbach
Sarah De Bruyne
Michael Unger
Mathias Wien
Rik Van de Walle

System architecture for semantic annotation and adaptation in content sharing environments

Published online: 5 June 2008

S. De Zutter (✉) · S. De Bruyne · R. Van de Walle
Department of Electronics and Information Systems, Multimedia Lab, Ghent University and Interdisciplinary Institute for Broadband Technology (IBBT), Gaston Crommenlaan 8 bus 201, 9050 Ledeberg-Ghent, Belgium
saar.dezutter@elis.ugent.be

M. Asbach · M. Unger · M. Wien
Institute of Communications Engineering (IENT), RWTH Aachen University, 52056 Aachen, Germany

Abstract This paper describes a system architecture, which enables the automatic semantic annotation and adaptation of multimedia content in context-aware content sharing environments. The discussed architecture is the result of research done in the EU FP6 IST INTERMEDIA project. Generating a common vision on user-centric multimedia services in shared content environments to provide users with content personalized to their user preferences and usage environment is one of the objectives of the project. The work presented in this paper describes how media formats with their related metadata are automatically annotated and dynamically adapted. Based on the architecture, a full-featured demonstrator is built.

Keywords Semantic annotation and adaptation · Description-driven adaptation · System architecture · Content sharing environment

1 Introduction

The heterogeneity of networks, devices, protocols, and multimedia formats continuously fosters the ever-growing need for novel multimedia services. This plethora endangers the development of dynamic interoperable multimedia facilities. Dealing with such a degree of complexity is a major technical and economical challenge, which affects network operators, content and service providers, and users. The architecture described in this paper addresses the delivery and consumption of adaptable multimedia content in a context-aware content sharing environment. The architecture facilitates the construction of user-centric multimedia services.

The research described in this paper is the result of research performed in the scope of the EU FP6 IST project Interactive Media with Personal Networked Devices (INTERMEDIA) [9]. One of the objectives is to generate a common vision on user-centric multimedia services in shared content environments to provide users with content personalized to their (semantic) user preferences and usage environments. In order to achieve this goal, we have specified the integrated components of a system architecture for automatic content annotation, dynamic content adaptation, and flexible content delivery and consumption. A demonstrator for semantic annotation and adaptation has been developed to illustrate the user-centric multimedia service concepts made possible by the proposed architecture. The resulting technology enhances the overall user experience by providing service quality customized to given semantic user preferences. Providing content for shared environments, which needs to be adapted to dynamic context information is a significant challenge. Unfortunately, most content collections lack any metadata for efficient browsing, retrieval, and adaptation. Manual annotation is very resource consuming, if not also technically unfeasible for the user. Therefore, to extract the semantics from the content, automatic content analysis and annotation are of imperative importance. Content adaptation should happen independently of the underlying data format and the underlying kind of adaptation executed on the content (e.g., spatial scalability or key frame selection). Content providers in such an environment benefit from storing the content only once, annotating it automatically, and (dynamically) adapting it to the needs of every individual user. The challenges, which were encountered by specifying this system architecture, are the following:

- Offering user-centric context-aware multimedia services, taking into account the characteristics of a content sharing environment and the evolving usage context, e.g., terminal constraints and semantic user preferences
- Support for automatic content annotation and support for streaming in content sharing environments
- Integrating the technologies for annotating content, semantic preferences descriptions, and structural descriptions of video streams into an extensible and interoperable architecture for format-agnostic content adaptation
- Technical validation of the developed tools by creating a real-time demonstrator that integrates all developed components

The remainder of this paper is organized as follows. First, the system requirements are given in Sect. 2. Then, Sect. 3 summarizes the different multimedia services supported by the architecture and describes the developed architecture. An in-depth discussion of the semantic annotation and adaptation techniques developed in the INTERMEDIA project can be found in Sect. 4. The semantic annotation and adaptation demonstrator, showing the use of the architecture in a real-life environment and the data flow of the user-centric multimedia services within the architecture are described in Sect. 5.

2 System requirements

The user interacting with the content sharing platform should be able to control the playback of content on a terminal and configure his or her user profile (including his semantic user preferences). Having content that needs to be shared and adapted according to the usage environment implies that the system architecture needs to support standardized multimedia data formats for content, such as audio, video, 3D, presentation level information, and structural descriptions. Standardized metadata formats also need to be used to describe the metadata of the available content. The preference is on ISO standards, like MPEG-4 [13], MPEG-7 [12], and MPEG-21 [3]. The annotation tools must be able to process the standardized multimedia data formats in the content sharing environment in order to generate annotations. These annotations are either of direct interest to the user or meant to guide other multimedia services, such as content adaptation. The output of the annotation tools should be in a pre-defined data format, such as the standardized MPEG-7 format. Adaptation tools provide the bridge between the (structure of the) available content, the annotations, and the user preferences as well as his or her current usage environment. This imposes several requirements. The adaptation tools must be able to dynamically work with the content and metadata formats provided by the content sharing environment. Since the goal of adaptation is to dynamically change properties of multimedia data to the user's current requirements, these tools must be able to obtain and process the (varying) usage environment and (semantic) user preferences, and reflect both of them in the adaptation decisions. The information exchange between annotation tools and adaptation tools thus requires precise API descriptions. Again the use of international standards (e.g., MPEG-21) is recommendable. For example, a user has indicated that he or she is most interested in the sports sections of news sequences. The adaptation tools should be able to not only semantically adapt the content stream to remove scenes of lower interest to the user but also to react to the environment of the user. For instance, a user might also be interested in traffic information when listening to news sequences in the car. Note that to support distributed adaptation in a content sharing environment, the adaptation tools need to be able to be spread across the network (for example by means of appropriate XML technologies). Since a content sharing environment aims at being deployable in streaming scenarios, the streaming architecture should support interaction with the user and support progressive streaming of adapted resources to the terminals.

3 System architecture for adaptable content sharing

The following section describes the different multimedia services supported in the content sharing environments we address. From a user's perspective, these services are the functionalities provided by our system architecture for annotation and adaptation.

3.1 Supported multimedia services

Figure 1 shows the relationships between the offered services. The content management service is responsible for the application logic involved in content consumption applications. This service processes content in the central content repository needed by the other services.

For adaptation as well as for content retrieval, a description of the available content is required. In content sharing environments, it cannot be assumed that all content is already suitably annotated or even annotated at all. However, it should be transparent to a user if annotations come from a service provider or are generated. Thus, automatic, non-interactive content analysis and description is started whenever new content is added to the central content repository.

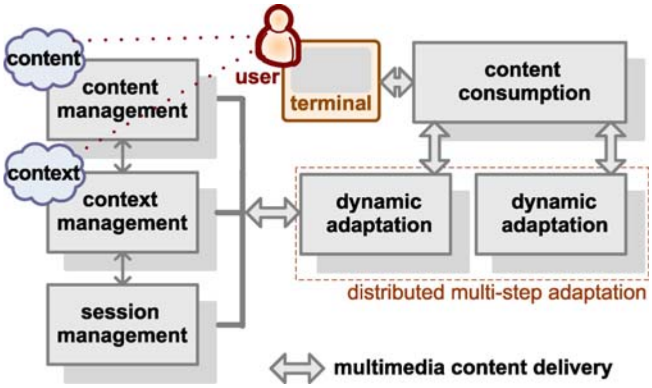

Fig. 1. Services supported by the system architecture

To enable universal multimedia access (UMA) [17] in a truly context-aware environment, there is a strong need for interoperable context management services. Therefore, generic inclusion of the context information, gathered by the context management service in the multimedia delivery and consumption chain, has proven to be of great importance [18]. Different kinds of context information can be collected at specific moments and stored by this service, for instance, user profile context information, such as the statically submitted semantic user preference for sport scenes. Usage environment information can also be collected by software upon change of certain characteristics. For example, when a user switches to another terminal, the screen resolution and multimedia codec of the new terminal are detected.

When the user's preferences and/or usage environment characteristics vary over time, the resource being consumed is probably no longer optimized to the new context information. A dynamic adaptation service for digital resources plays an important role here. Over time, this service dynamically adapts the streamed resource to the varying context information during the streaming phase [14] triggered by the context management service when context information changes. Different kinds of resource adaptation exist (e.g., description-driven semantic adaptation (see Sect. 4.3), resource conversion, presentation adaptation, transmoding, and transcoding) and can be done in multiple distributed steps. For instance, the multimedia resource suited for the consuming multimedia codec is selected. Based on the semantic preference of the user, highlights of the news can be selected. Then, this selected content is adapted to the screen resolution of the consuming terminal. To cope with fluctuations in the available bandwidth, the quality of the streamed resource is dynamically adapted achieving a suited bit rate. The adaptation service is a vital service in a user-centric content sharing environment to enable the user to consume content located on a central content repository by adapting it to a format that is on top of being consumable on the user terminal, also optimized to the user preferences and usage environment.

Finally, the content delivery service provides delivery of this content to a wide range of terminals connected to different networks. All communication involved in consuming and streaming multimedia resources is done by this service. The content is accessed by the server making use of the delivery service and adapted to the specific context information, using adaptation services. Subsequently, the most suited (adapted) multimedia resource is streamed to the client terminal. Note that a session management service is especially useful for user-centric architectures, wherein the usage environment of the user is important. By keeping track of active sessions, context information is linked to users.

3.2 Architecture

In this section, the mapping between the multimedia services to their architectural system counterparts is described. Figure 2 illustrates the conceptual system components that are part of a content sharing environment which supports semantic content annotation and adaptation. First of all, content sharing systems need to efficiently manage (scalable) content in terms of delivering it in appropriate formats to different terminals. Scalable coding formats certainly need to be taken into account here to enable the extraction of multiple versions of the same resource without the need for a complete recoding process. Due to the diversity of coding formats in the content sharing environment, using format-independent adaptation tools considerably improves efficiency. These tools adapt content independent of the underlying coding format. Context-aware systems provide information on the usage environments and annotation tools provide descriptions of the content streams to be used to control the adaptation of the content. When adaptation needs to be supported in a distributed content sharing environment, the streaming server

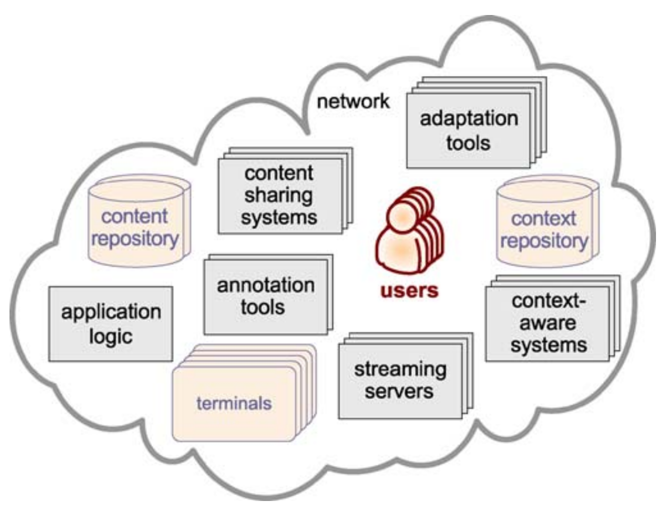

Fig. 2. System architecture for annotation and adaptation

is also an important component. From a system's point of view, the conceptual components of a content sharing architecture that supports content annotation and adaptation are:

– A (pool of) content sharing system(s) having links to central content repositories containing content coded with (scalable) coding formats for still images, audio resources, and video resources as well as links to central context repositories with metadata about this content.
– A (pool of) context-aware system(s) to collect, aggregate, and manage the context information from the usage environment in the context repositories. Examples of context collection tools are sensor-triggered tools for probing the available bandwidth and off-line semantic user preference profiling tools.
– An (pool of) annotation tool(s) for feature extraction, content and scene classification, and content annotation steered by applications and/or by users.
– A (pool of) distributed adaptation tool(s) supporting format-specific adaptation tools (e.g., transcoders) and format-agnostic adaptation techniques (e.g., based on MPEG-21 DIA BSDL or gBS Schema [3]). The benefit of such generic adaptation engines is that the underlying implementation does not have to be updated to support a new coding format, as is further elaborated on in Sect. 4.3.
– A (pool of) distributed streaming server(s) to disperse the streaming work load over the network.
– Application logic which serves as a layer between all components for selecting streaming servers and adaptation engines, accessing the content sharing systems, controlling the annotation tools and context-aware adaptation decision taking processes, for communication between terminals and the server, for interaction with the user, etc.
– Terminals which are offered streamed content that is adapted to the usage environment through the content sharing system. To increase scalability of the content sharing environment these architectural components should be modular so that they can be dynamically distributed across a managed network.

4 Semantic annotation and adaptation

4.1 Principles

In order to efficiently deal with the heterogeneity in the usage environments, the content sharing system needs to rely on efficient annotation and adaptation strategies. This section focuses on semantic content adaptation (described in Sect. 4.3), being the extraction of desired content segments based on the (semantic) user preferences, thereby linking the automatically generated high-level structure descriptions of the content to the semantic annotations added by the automatic content analysis (described in Sect. 4.2) and the user preferences. The high-level description of our semantic annotation and adaptation scenario is as follows. First, when new video sequences are added to the content sharing system, they are analyzed, annotated, and described.

1. Shot boundary detection and scene classification from pre-trained scene types like anchor person, weather, and sports are performed to generate semantically meaningful annotations.
2. High-level structural content descriptions are generated from the content.

Then, when content is requested to be consumed by the user:

3. The current user context and semantic preferences are correlated with the semantically annotated classifications by the adaptation process.
4. Based on this correlated metadata and the structural descriptions of the content, the content is adapted and streamed to the user.

4.2 Semantic annotation

4.2.1 Shot boundary detection

The most valuable metadata available for multimedia streams is their temporal structure. However, if the temporal information is not present or not detailed enough, it has to be generated from the multimedia itself. Shot boundary detection is the first step in multimedia annotation and has been under investigation for multimedia analysis extensively [6].

Here, shot boundaries are detected based on color histograms in HSV color space. As depicted in Fig. 3 these color histograms are averaged over a couple of frames. This way, small jumps in histogram entries are smoothed and only non-transient changes result in a jump in histogram differences that indicate a shot boundary (see Fig. 4).

In the literature, evaluation results are hard to compare because of varying test sets, different ground truth, and non-standard or even inappropriate evaluation metrics. Given the character of the shot boundary detection task, it seems advisable to present results in recall r and precision p rather than accuracy or ROC curves. To judge the quality of a given operating point, the $F1$ score is calculated:

$$F1 = \frac{2 \times p \times r}{p + r}. \quad (1)$$

An evaluation of the performance of our algorithm on a diverse set of test videos is given in Table 1. For this evaluation, the same, fixed threshold, number of histogram bins, and number of frames contributing to the averages are chosen.

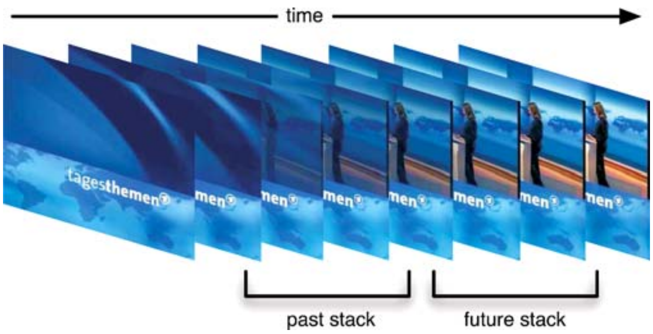

Fig. 3. Color histograms are averaged over a stack of past and future frames before comparison

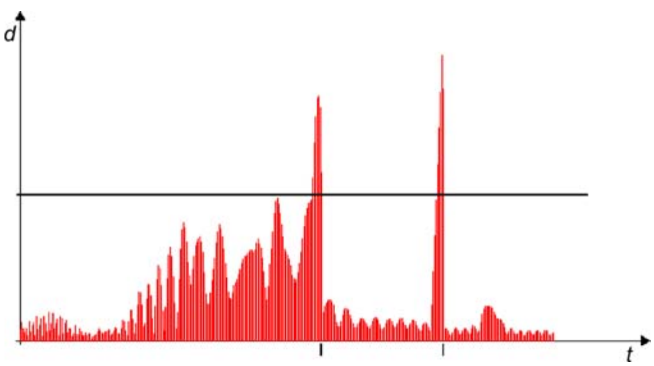

Fig. 4. Typical time series of color histogram differences d showing a dissolve and a hard cut

For every shot, a representative frame (key frame) is chosen by selecting the frame that has the smallest difference to the average histogram of the shot. Such a key frame represents the whole shot for user interaction and is typically used in subsequent content description tasks.

In addition to the abrupt shot boundary detection, a long-term comparison with the start of the current shot allows for the detection of gradual changes resulting from transition effects like wipes or dissolves. Depending on the speed of a transition, it is either marked as a shot boundary, or it just causes the generation of an additional key frame for the current shot. This allows for the description of shots with strong camera movement or a multitude of small changes.

4.2.2 Scene classification

The detection of scene types of course requires training data for every category. In our case, for each scene type, several key frames obtained from shot boundary detection are collected. These images are transformed into a representation that combines histogram-based descriptions with pixel-wise image intensities and variances. A clustering step groups visually similar key frames that represent subcategories of a given scene type.

Table 1. Evaluation results for shot boundary detection

Sequence	Recall	Precision	$F1$
News	0.94	0.91	0.93
Matrix	0.50	0.48	0.49
Pixar	0.82	0.92	0.87

Training. First, for each image $\boldsymbol{I}_\mathrm{t}$ a color histogram $\boldsymbol{H}_\mathrm{t}$ of the training data set is calculated, where $\boldsymbol{H}_\mathrm{t}$ is composed of three histograms, one for each color. Each histogram contains $x = 1, \ldots, 256$ entries $\boldsymbol{H}_\mathrm{t}(x) = [H_{\mathrm{r,t}}(x), H_{\mathrm{g,t}}(x), H_{\mathrm{b,t}}(x)]^T$ according to the 8 bit color range of a pixel in RGB color space.

Assuming that a list of clusters is already available, a new key frame may either be grouped with an existing cluster or is considered as a new cluster. For this purpose, the dissimilarity between the histogram of the new image and the histograms of the clusters within the feature space is derived. The dissimilarity d between the histogram of the new image $\boldsymbol{H}_\mathrm{n}$ and one histogram of a feature point $\boldsymbol{H}_\mathrm{f}$ is calculated by:

$$d = \sum_{c=\mathrm{r,g,b}} \sum_{x=1}^{256} |H_{c,\mathrm{n}}(x) - H_{c,\mathrm{f}}(x)|. \tag{2}$$

If d is below a certain threshold, the image is merged with the most similar cluster and the average histograms for that cluster are updated.

After all training images are processed, for each cluster i the histogram $\boldsymbol{H}_i$, the mean image $\boldsymbol{M}_i$, and a variance image $\boldsymbol{V}_i$ are calculated. An example of a mean and a variance image is given in Fig. 5.

Classification. During the classification phase, every image of a video is analyzed, whether it belongs to a trained class or not. For this purpose, an empirical decision function is used. The decision function utilizes the dissimilarity value d between the histogram of a cluster and the histogram of the current frame (see Eq. 2). In addition, the weighted sum of squared differences s_i between the mean color image $\boldsymbol{M}_i$ of a cluster i and the current

Fig. 5. Mean and variance image of a single cluster of an anchor person scene from news casts

frame $\boldsymbol{I}$ is calculated, with the clusters' pixel-wise variances being used as a weight:

$$s_i = \frac{1}{3wh} \sum_{c=\mathrm{r,g,b}} \sum_{x=1}^{w} \sum_{y=1}^{h} \boldsymbol{V}_{c,i}(x, y) \cdot [\boldsymbol{I}_c(x, y) - \boldsymbol{M}_{c,i}(x, y)]^2, \quad (3)$$

where w and h correspond to the width and height of the images. An image does not belong to a class, if either $d > t_1$, $s > t_2$, or $s \cdot d > t_3$ is satisfied. The thresholds t_1, t_2, and t_3 were empirically determined, using the training data of all scene types that have to be classified.

4.2.3 Content annotation

Annotations sometimes are of direct interest to the user. He might, for example, want to see a table of contents generated from the temporal segmentation of a video clip. This means that all annotations should have representations that are easy to present to a human user. On the other hand, annotations are generated to guide the adaptation processes. Thus they need to be machine-readable and should be easy to parse.

MPEG-7 provides all the necessary tools to describe multimedia content, including but not restricted to the temporal structure of bitstreams. The complexity of the MPEG-7 standard however allows for files that are arbitrarily complex themselves. To allow a less complex implementation, we have defined a restricted subset of MPEG-7 that allows for storing segment descriptions in a flat, non-nested line-up. Also, one of the key insights of research on multimedia adaptation is that the most and presumably the best tools to process knowledge exist for text, compared to other kinds of data. As a consequence, we have decided to annotate with text tags only. Where MPEG-7 provides numerical descriptors for structural information like shape, color, or facial properties, we only store the result of a detection or classification process in a textual form.

4.3 Semantic adaptation

The supported coding formats and the implemented adaptation processes are vital factors in a powerful content sharing environment. Especially the use of scalable formats and format-agnostic adaptation engines considerably enhances efficiency in managing content in such content sharing environments. The use of MPEG-21 DIA gBS Schema-driven content adaptation enables the creation of these format-agnostic adaptation engines, since the engines operate at the XML level and do not have to be aware of the underlying coding format [5]. The additional advantage of using XML descriptions is the integration with other metadata standards, such as MPEG-7.

Since using this XML-driven adaptation technique has certain consequences for the content sharing environment, a concise description is provided here. The concept of XML-driven adaptation is based on the transformation of XML descriptions representing coded content bitstreams. This implies three basic steps, as can be seen in Fig. 6. First of all, high-level XML descriptions (in this case generic Bitstream Descriptions (gBSDs)) of the content bitstreams available in the content sharing system need to be automatically generated and are stored in the content repository. For details on the technicalities of MPEG-21 gBS Schema, the reader is referred to [3]. However, it is important to note that the use of

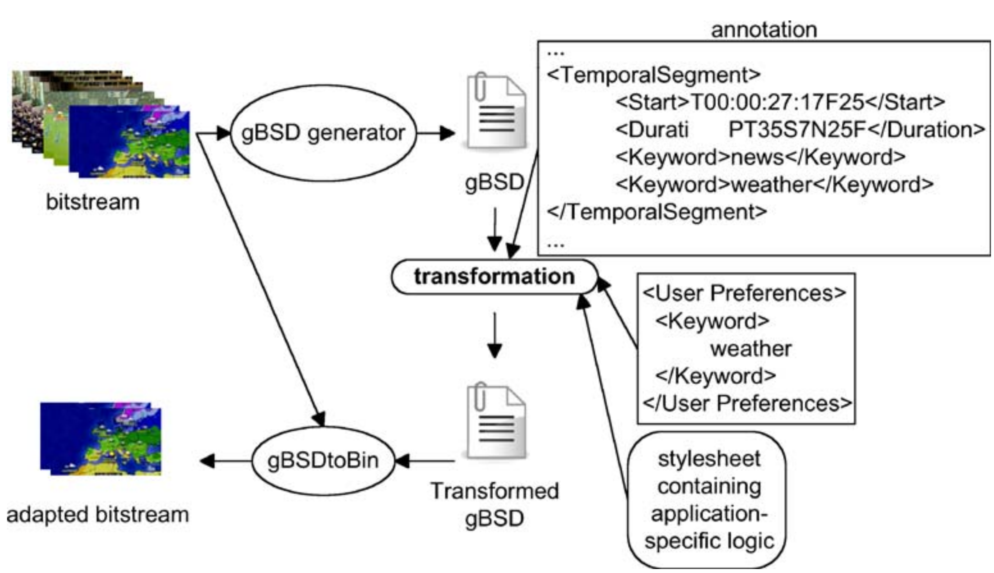

Fig. 6. Semantic gBSD-based adaptation

markers in gBSDs allows for the creation of timing information in the hierarchical structure of the bitstream. Exactly those markers are later on used in the semantic adaptation process.

Next, the gBSD has to be transformed; the adaptation of this XML description takes the usage environment and the annotation that is available through the context repositories into account. Content adaptation enables the personalization of the content according to the preferences of a user. However, special attention needs to be paid to the extraction of the desired content as the adapted bitstream needs to remain compliant with the corresponding specification. To this end, we use the algorithm proposed in [5]. Different approaches exist to transform XML descriptions into adapted XML descriptions, for instance by means of programming languages (e.g., Java or C++) and an XML parser. A more efficient approach is to use XML style sheets and an engine to interpret these style sheets (e.g., eXtensible Stylesheet Language Transformations (XSLT) [4] and Streaming Transformations for XML (STX) [15]). The main advantage of the latter approach is the possibility to build generic software modules to perform the adaptation of the XML descriptions. Note that only event-based models for XML parsers are suitable for use in resource-constrained and streaming environments. Moreover, only when using the format-agnostic STX technology, the adaptation engine is fully format-agnostic. In order for the annotation tools and the adaptation engine to work together, the STX style sheets need to contain meaningful application logic to be used in the adaptation process to apply the necessary XML transformations to the gBSD descriptions based on the information provided by the annotation tools.

The last step is the creation of an adapted content bitstream based on the original coded bitstream and the adapted gBSD description. This translation from the XML domain to the binary domain happens through the gBSDtoBin process, which is normatively defined in the MPEG-21 DIA specification [3].

5 Demonstrator

5.1 Data flow

This section focuses on the interaction between the different components in the INTERMEDIA semantic annotation and adaptation demonstrator and illustrates how the different components of the client-server architecture are linked to each other. The demonstrator that implements the system architecture for semantic annotation and adaptation is depicted in Fig. 7a. Figure 7b shows the different steps in the data flow of the content delivery and consumption chain of the demonstrator, starting with user-centric context and content management. The user can indicate his or her semantic preferences, which are recorded and stored by the context management tool (1) in the context repository (2). When MPEG-4 content is added to the content sharing environment, several actions are initiated by the content management tool. First, the content management tool invokes the content annotation tool chain (3). Shot boundary detection is performed as described in

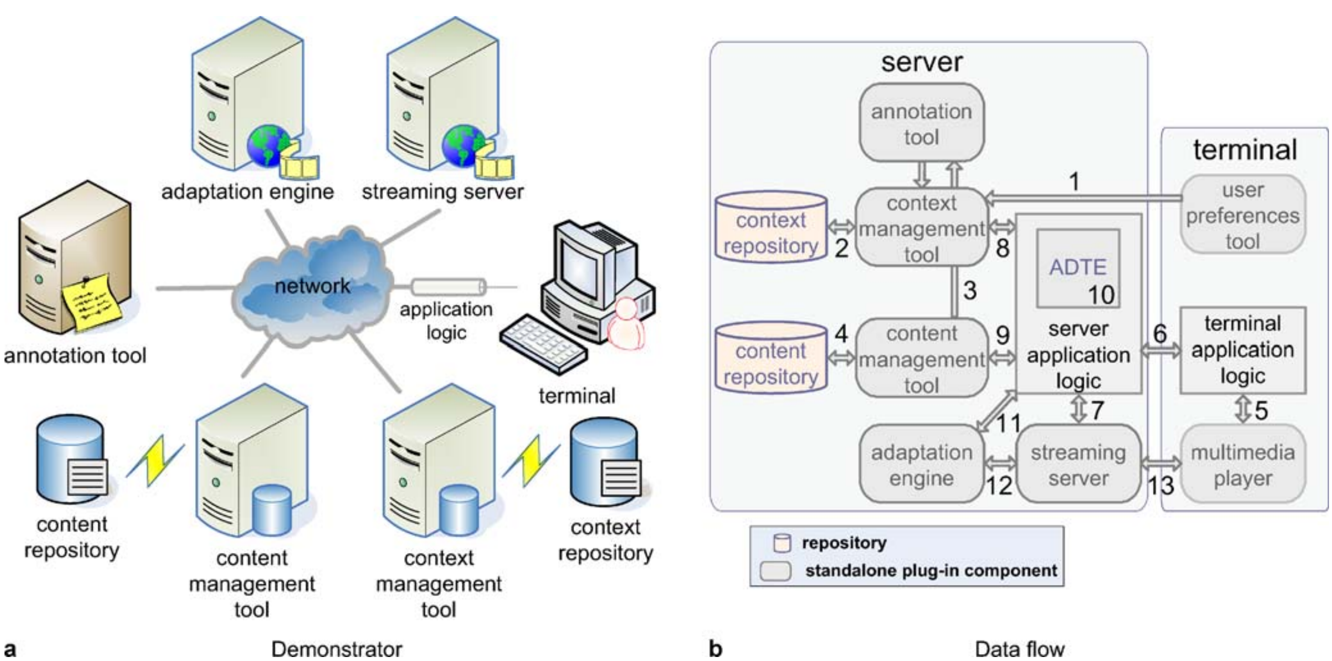

Fig. 7a,b. Demonstrator developing the system architecture for semantic annotation and adaptation

Sect. 4.2.1 to create an initial content description. From every temporal segment, a representative image (i.e., a key frame) is extracted. These images together with a list of trained classifiers is passed to the scene classification tool (see Sect. 4.2.2). For every positive classification result, a text tag with the name of the classifier that reported a hit is added to the respective temporal segment of the content description. At the same time, as explained in Sect. 4.3, the content repository is extended to include a gBSD of this coded video stream (4).

The user interaction for controlling the playback of content with the terminal is conducted via the terminal application logic communicating with the multimedia player (5). When a user requests the terminal application logic on the terminal to consume content, the request is forwarded to the server side (6). Then, an appropriate streaming server is selected based on the current server load and is presented to the client terminal (7). Thereafter, the server application logic will fetch the semantically annotated metadata about the requested content and the user preferences of the user requesting the content, which is stored in the context repository (8). The structural information describing the requested content is also retrieved from the content repository by the server application logic (9). The adaptation parameters are defined by matching the user preferences of the requesting user and the semantically annotated metadata of the content in the internal adaptation decision taking engine (ADTE) server component (10). Subsequently, the adaptation engine is initiated based on the acquired adaptation parameters and by accessing the content repository (11). Finally, once the adaptation is started by the server application logic, the streaming server can progressively read the adapted content from the adaptation engine (12) and send it to the terminal for playback (13).

5.2 Implementation

The content repository contains MPEG-4 file containers which need to be annotated using trained scene types. In addition, structural metadata describing the bitstreams, being gBSDs, are present. The context repository contains semantic metadata of the user (in this demonstrator this is a proprietary XML structure for user preferences) and semantic metadata of the content, i.e., metadata resulting from scene classification of a particular video sequence. The annotation tools are implemented as a mix of Python and C++ code that relies on the OpenCV computer vision library [2]. Initial versions of these annotation tools are now available as Open Source software on SourceForge [1]. More details about the implementation of the adaptation engine can be found in Sect. 4.3. The streaming server makes use of the Live555 Streaming Media [11] server application for its communication with the other components and the terminal. This streaming server utilizes RTSP [8] for interaction with the user consuming the content and RTP [7] for streaming the adapted content to the terminals. The terminal embeds the Osmo4 Player [10] for playback of the streamed adapted video streams. A more in-depth discussion of the multimedia framework used here for the content delivery and consumption chain is described in [16].

6 Conclusions

By combining efforts from different research domains, we have created a system architecture for a content sharing environment supporting automatic content analysis and format-agnostic adaptation. The scheme presented in this paper bridges the gap between the metadata added by annotation tools and format-agnostic semantic content adaptation by making use of MPEG-21 gBS Schema. As the proposed hierarchical structure of the adaptation process does not depend on the type of content nor the kind of adaptation that is performed on the content, the offered adaptation service does not need to be updated whenever a new data format is ready to be consumed. By integrating user-centric context-aware multimedia services using standardized data and metadata formats and transport and delivery mechanisms, we have built an extensible and interoperable system architecture for automatic semantic content annotation and adaptation ready for use in content sharing environments.

Acknowledgement The research activities that have been described in this paper were funded by Ghent University, the Interdisciplinary Institute for Broadband Technology (IBBT), the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT), the Fund for Scientific Research-Flanders (FWOFlanders), the Belgian Federal Science Policy Office (BFSPO), and the European Union (within the framework of the NoE INTERMEDIA, IST-038419).

References

1. Asbach, M., Unger, M.: INTERMEDIA Multimedia Tools. http://sourceforge.net/projects/intermedia (2008). Accessed 2008
2. Bradski, G.: The OpenCV Library. Dr. Dobb's J. Software Tools (2000)
3. Burnett, I., Pereira, F., Van de Walle, R., Koenen, R. (eds.): The MPEG-21 Book. Wiley, New York (2006)
4. Clark, J.: XSL Transformations (XSLT) version 1.0. W3C Recommendation, W3C (1999)
5. De Bruyne, S., De Schrijver, D., De Neve, W., Van Deursen, D., Van de Walle, R.: Enhanced shot-based video adaptation using MPEG-21 generic bitstream syntax schema. In: Proceedings of the 1st Symposium on Computational

Intelligence in Image and Signal Processing, pp. 380–385. IEEE Computer Society, Washington, DC (2007)
6. Hanjalic, A.: Shot-boundary detection: unraveled and resolved? IEEE Trans. Circuits Syst. Video Technol. **12**(2), 90–105 (2002)
7. Internet Engineering Task Force (IETF): RFC 1889: RTP: A Transport Protocol for Real-Time Applications (rtp). IETF RFC, IETF. http://www.ietf.org/rfc/rfc1889.txt (1996). Accessed 2008
8. Internet Engineering Task Force (IETF): RFC 2326: RTSP: Real-Time Streaming Protocol (RTSP). IETF RFC, IETF. http://www.ietf.org/rfc/rfc2326.txt (1998). Accessed 2008
9. INTERMEDIA: Interactive Media with Personal Networked Devices (ist-1-38419) (2008). European Sixth Framework Programme (FP6) IST NoE co-funded project http://intermedia.miralab.unige.ch/. Accessed 2008
10. Le Feuvre, J., Concolato, C., Moissinac, J.C.: GPAC: Open Source Multimedia Framework. In: MULTIMEDIA '07: Proceedings of the 15th International Conference on Multimedia, pp. 1009–1012. ACM, New York (2007)
11. Live555 Streaming Media: Live555 Streaming Media. http://www.live555.com/liveMedia/ (2007). Accessed 2008
12. Manjunath, B., Salembier, P., Sikora, T.: Introduction to MPEG-7: Multimedia Content Description Interface. Wiley, New York (2003)
13. Pereira, F., Ebrahimi, T. (eds.): The MPEG-4 Book. Prentice Hall, New York (2002)
14. Ransburg, M., Cazoulat, R., Pellan, B., Concolato, C., De Zutter, S., Poppe, C., Hutter, A., Hellwagner, H., Van de Walle, R.: Dynamic and distributed adaptation of scalable multimedia content in a context-aware environment. In: MOBIMEDIA '06: Proceedings of the 2nd International Mobile Multimedia Communications Conference. ACM, New York (2006)
15. SourceForge: Streaming Transformations for XML (STX). http://stx.sourceforge.net/documents/spec-stx-20070427.html. Working Draft (2006). Accessed 2007
16. Van Deursen, D., De Bruyne, S., Van Lancker, W., De Neve, W., De Schrijver, D., Hellwagner, H., Van de Walle, R.: MuMiVA: a multimedia delivery platform using format-agnostic, XML-driven content adaptation. In: Proceedings of the 9th International Symposium on Multimedia, pp. 131–138. IEEE Computer Society, Washington, DC (2007)
17. Vetro, A., Christopoulos, C., Ebrahimi, T.: Universal multimedia access. IEEE Signal Process. Mag. **20**(2), 16 (2003)
18. Zimmerman, A., Specht, M., Lorenz, A.: Personalization and context management. User Model User Adapted Interact. **15**(3–4), 275–302 (2005)






S. De Zutter received her Master's degree in Computer Science from Ghent University, Belgium in 2002. Since 2003, she has worked as a Ph.D. researcher at Multimedia Lab, Ghent University, IMEC, and IBBT. Her research topics include dynamic and distributed adaptation and processing of multimedia content in context-aware environments and session transfer in mobile applications. Most of her research was performed within the European projects DANAE and INTERMEDIA. She actively participated in standardization activities in the Moving Picture Experts Group, which resulted in the editorship of five MPEG-21 standards. Saar De Zutter is (co-)author of several international conference and journal papers.

M. Asbach was born in 1976 in Neuss, Germany. He received the Dipl.-Ing. degree in Electrical Engineering from RWTH Aachen University, Germany, in 2004. Since then, he has been working towards a Ph.D. degree at the Institute of Communications Engineering, RWTH Aachen University. His research interests are pattern recognition, image processing, and computer vision, targeting multimedia content analysis.

S. De Bruyne received her Master's degree in Computer Science from Ghent University, Belgium in 2005. Since then, she has worked as a Ph.D. researcher at Multimedia Lab, Ghent University, and IBBT. Her research interests are in the areas of multimedia signal processing, content-based multimedia analysis, video coding technologies, and content adaptation. Part of her research was performed within the European project INTERMEDIA.

M. Unger was born in 1977 in Bad Neuenahr, Germany. He received the Dipl.-Ing. degree in Electrical Engineering from RWTH Aachen University, Germany, in 2004. Since then, he has been working towards a Ph.D. degree at the Institute of Communications Engineering, RWTH Aachen University. His research interests are 3D reconstruction, image processing, and computer vision.

M. Wien received his Dipl.-Ing. and Dr.-Ing. degrees from RWTH Aachen University, Germany, in 1997 and 2004, respectively. From 1997 until 2004, he worked towards his 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 and 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.

R. Van de Walle received his M.Sc. and Ph.D. degrees in Engineering from Ghent University, Belgium in 1994 and 1998, respectively. After a visiting scholarship at the University of Arizona (Tucson, USA), he returned to Ghent. In 2001, he became a professor at the Department of Electronics and Information Systems, and founded the Multimedia Lab. He has been editor of several MPEG specifications, and has been involved in the organization of several international conferences. His current research interests include multimedia content delivery, presentation and archiving, coding and description of multimedia data, content adaptation, and interactive (mobile) multimedia applications. Multimedia Lab is one of the partners of the Interdisciplinary Institute for Broadband Technology (IBBT).