ARTICLE IN PRESS

Available online at www.sciencedirect.com
SCIENCE DIRECT®

Computer Communications xxx (2006) xxx–xxx

computer communications

www.elsevier.com/locate/comcom

MPEG-21 digital items to support integration of heterogeneous multimedia content

Kostas Karpouzis [a,*], Ilias Maglogiannis [b], Emmanuel Papaioannou [c], Dimitrios Vergados [b], Angelos Rouskas [b]

[a] *Image, Video, and Multimedia Systems Laboratory, National Technical University of Athens, 9, Heroon Polytechniou street, 157 80 Athens, Greece*
[b] *Department of Information and Communication Systems Engineering, University of the Aegean, 83200 Karlovasi, Greece*
[c] *Intracom S.A., Markopoulo Av., 19002 Peania, Greece*

Abstract

The MELISA system is a distributed platform for multi-platform sports content broadcasting, providing end users with a wide range of real-time interactive services during the sport event, such as statistics, visual aids or enhancements, betting, and user- and context-specific advertisements. In this paper, we present the revamped design of the complete system and the implementation of a middleware entity utilizing concepts present in the emerging MPEG-21 framework. More specifically, all multimedia content is packaged in a self-contained "digital item", containing both the binary information (video, graphics, etc.) required for the playback, as well as structured representations of the different entities that can handle this item and the actions they can perform on it. This module essentially stands between the different components of the integrated content preparation system, thereby not disrupting their original functionality at all; additional tweaks are performed in the receiver sides, as well, to ensure that the additional information and provisions are respected. The outcome of this design upgrade is that IPR issues are dealt with successfully, both with respect to the content itself and to the functionality of the subscription levels; in addition to this, end users can be presented with personalized forms of the final content, e.g., viewing in-play virtual advertisements that match their shopping habits and preferences, thus enhancing the viewing experience and creating more revenue opportunities via targeted advertisements.

Keywords: Heterogeneous content adaptation; Metadata; MPEG-21; User modelling; MPEG-4

1. Introduction

In the framework of digital TV, the new generation of viewers is currently being confronted and becoming acquainted with a series of technological developments in the realm of consumer electronics and gaming that raise their expectations for similar advances in TV broadcasts. Domains characterized by inherent dynamics, such as sports broadcasting, make these expectations even higher; thus, broadcasting corporations and organizations need to preserve or build up their competitive advantage, seeking new ways of creating and presenting enhanced content to their customers. From the Mobile Devices point of view, third-generation transmission technology, like the Universal Mobile Telecommunication System (UMTS), will greatly enhance the multimedia potential of mobile phones. The investments committed to deploying 3G systems are significant, pushing the need to offer innovative services over 3G networks in order to facilitate wide take up of the new technology by the users.

Sports events continue to attract interest and are among the most popular media attractions in the world today. Correspondingly, multimedia applications developed for them have a huge potential market impact. These applications include the provision of enhanced visual and text con-

* Corresponding author. Tel.: +30 210 772 3037; fax: +30 210 772 2492.
E-mail addresses: kkarpou@cs.ntua.gr (K. Karpouzis), imaglo@aegean.gr (I. Maglogiannis), paem@intracom.gr (E. Papaioannou), dvergad@aegean.gr (D. Vergados), arouskas@aegean.gr (A. Rouskas).

0140-3664/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2005.11.019

tent, dynamic content, such as viewing enhancements and advertisements that interacts with both the user and the audiovisual content itself and real-time bet placement via an intuitive interface. This framework involves two types of user terminals, namely Set-Top-Boxes (STB) and Personal Digital Assistants (PDA), which are characterized by inherent diversity with respect to their capabilities of receiving, processing and displaying multimedia content; regarding the actual content itself, since the integrated system handles, encodes and delivers multimedia content coming from different vendors, the respective Intellectual Property Rights (IPR) must be retained throughout the complete process. These two additional requirements can be dealt with via the inclusion of concepts presented within the emerging MPEG-21 framework, introducing additional middleware components which mediate the production, transmission and playback phases; essentially, this means that the established production process between the different content providers is not altered in any way, a feature which is crucial since all content providers already work on established processes and patterns. This is illustrated in Fig. 1, which shows an overview of the higher-level information flow, between the different system components; here, the components of the integrated system that undertake the task of collecting, packaging, and delivering the multimedia data have been collectively enhanced to cater for the aforementioned provisions in the final content that is transmitted to the end-user and stored in the system repository. When compared to the initial sender-side architecture, which is based on established multi-vendor and multi-platforms publishing system, it becomes apparent that this process is not reworked; instead, parts of the middleware software components have been upgraded as needed.

In this paper, we present an extensive overview of the authoring process in the MELISA framework and concentrate in the ways an MPEG-21-compliant middleware revamps this process to cater for the additional content and presentation requirements. This paper is organized as follows: in Section 2, we provide a general overview of the developed framework and the workflow of the authoring process. In Section 3, we detail the authoring framework for the integration of the content from the individual providers into the content package, while Section 4 focuses on the MPEG-21 concepts utilized in the middleware implementation. Finally, Section 5 concludes the paper.

2. Architecture of the MELISA integrated system

MELISA is an acronym for *Multi-Platform e-Publishing for Leisure and Interactive Sports Advertising*. The system architecture consists of two separate systems: the sender side for content preparation and management; and the receiver side that performs content reception and manipulation, and handles the interactive components (a more detailed description of the overall process-level architecture system is provided in [1]). The content preparation process in the sender side includes preparing the advertising content, preparing the lottery event to be offered during a broadcast, and designing descriptive templates for the display of enhanced content at transmission time. Content

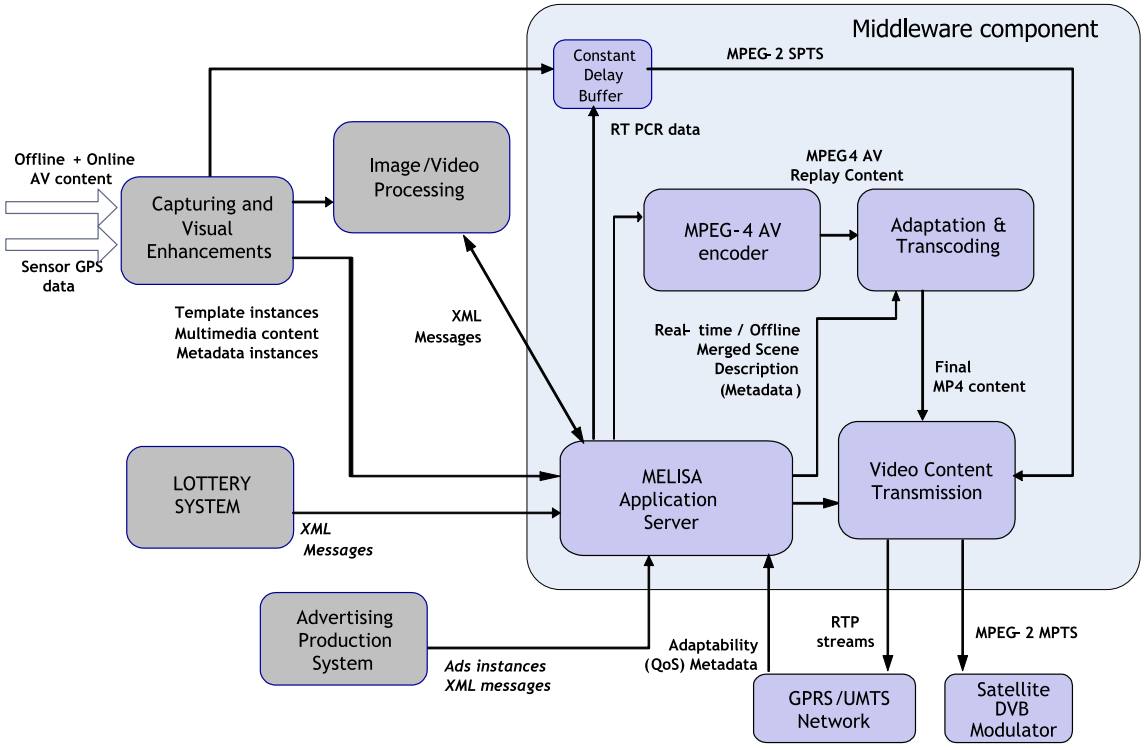

Fig. 1. Overview of the established content production and delivery architecture and the function of the MPEG-21 middleware.

adaptation to the user terminal (STB or PDA) is performed at the server side so that the content is created once, but adapted according to the targeted network and terminal prior to transmission to allow for efficient display and manipulation on the client side.

The overall architecture of the system at the sender side is given in Fig. 2. There are three major sub-systems grouped by numbers (1), (2), (3) in the figure:

- Editing and Workflow (1).
- Service Management (2).
- Multi-Platform Publishing (3).

Fig. 3 illustrates the Editing and Workflow sub-system. The sub-tasks in this unit include filming of the environment, offline scheduling of the event, preparation of enhanced content templates, and set-up of a repository structure corresponding to the schedule of the events with some basic metadata (name of athlete, nationality, etc.). Apart from these offline processing tasks, this sub-module also involves online authoring at transmission time, post-production, and workflow control tasks.

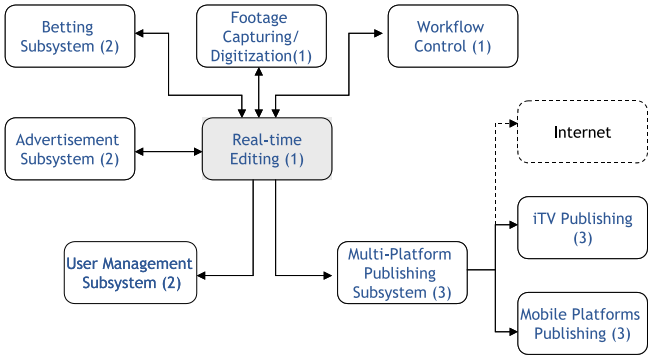

Fig. 2. A process-level schema of the sender architecture.

During each sport event, online capturing takes place continuously to avoid content and synch data loss when an enhanced replay is requested by the producer/director. A technician operates the capturing software, in accordance to the compiled schedule. The data captured are stored in the central production repository to be further used by the Visual Enhancements Engine during the template instantiation. When the director chooses to produce a replay, image/video processing tasks such as tracking are initiated. The resulting audiovisual content is fed into an MPEG-4 encoder. In addition to this, an operator driven Event Annotation System manages the real-time annotation of live data, and the triggering of events when required, e.g., when a goal score is recorded, the necessary scene updates are triggered automatically. This module works in close connection with the Visual Enhancements Engine.

The Service Management sub-system (2) takes care of the betting and advertising aspects. The described system offers the possibility of in-play betting, as a result of the developments taking place during an event. Such bets have to be approved by the betting sub-system administrator and have a limited "life cycle", since they are linked with specific parts or developments of a sports event.

Finally, as shown in Fig. 4, the Multi-Platform Publishing sub-system presents the final stage of tasks and modules during content production and publishing at the sender. All online and off-line content prepared by the other sub-systems are forwarded to the Information Merging Unit for the encoding and adaptation processes. The content is adapted to the different terminals and transmission networks targeted by the system (STB or PDA) and then delivered via the respective transmission channels. In the case of DVB-S broadcast, the video is streamed in MPEG-4 over an MPEG-2 Transport Stream (TS) The video resolution is then reduced to fit the lower transmission and playback capabilities of mobile terminals. Since the targeted receiver architectures offer different degrees

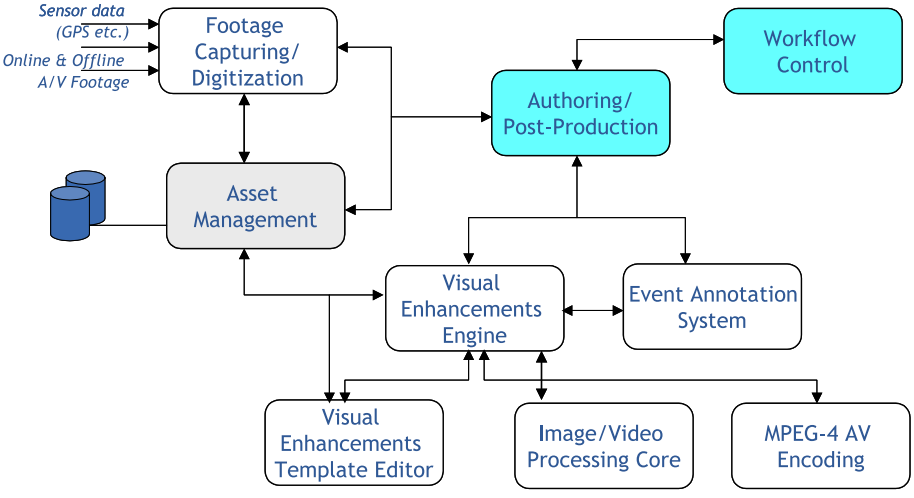

Fig. 3. A process-level schema of the Editing and Workflow sub-system (sub-system 1).

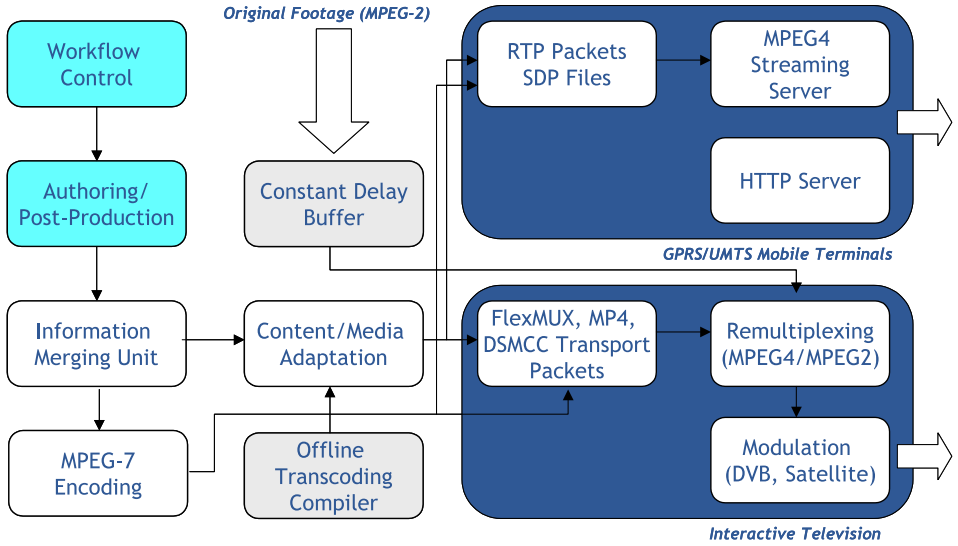

Fig. 4. An abstract schema of the Multi-Platform Publishing sub-system (sub-system 3).

of media delivery, interactivity, and responsiveness, it is essential to break down both the captured and synthesized material to match the relevant device. The publishing sub-system prepares different versions of the content for delivery.

An important part of the content preparation process is the template editing procedure. One can differentiate the templates into three categories, that of the visual enhancements, which consists of 2D templates, betting templates, and the advertisements templates. Templates are defined to merge all contents together, along with appropriately defined transformation stylesheets.

3. MPEG-4 rich media generation

3.1. Introduction

The process of authoring visual enhancements consists of two separate steps: offline and online authoring. Offline authoring takes place before the start of the sport event. It consists of creating an initial scene for the presentation. Online authoring consists of generating and transmitting real-time encoded visual enhancements during the event.

In our approach, the visual enhancements are encoded into MPEG-4 BIFS [2]. BIFS stands for Binary Format for Scenes. It is a binary representation of the spatio-temporal positioning of audiovisual objects and their behavior in response to interaction. Also, such as the VRML language [3] from which it has been derived, the BIFS standard allows to describe various 2D/3D vector graphics [4,5]. Central to the MPEG-4 framework is the concept of scene. A scene can be defined as *what the user sees and hears* [6]. It can be represented by a scene tree that describes the spatio-temporal relations of the different media objects to be displayed (i.e., video, audio, graphics, etc).

In the described system, BIFS are authored with the XMT (eXtensible MPEG-4 Textual format) language [7]. One characteristic of the BIFS standard is that it allows for a scene to be updated at different points in time. This is done by decoding what are called *BIFS Updates*. There are two mechanisms for updating or changing a scene. While BIFS-Anim Updates are used for updating parametric animations, BIFS-Commands Updates allow changing more general types of parameters in the scene. BIFS-Commands Updates, and more specifically the *Replace* Command (by distinction to the two other types of BIFS-Commands, *Insert* and *Delete*) are utilized in the system. This permits replacing various elements from the scene tree, from a complete node to a value in a multiple field, e.g., changing the color of a rectangle node, or changing the whole rectangle into a circle. BIFS Updates can be contained in the same file as the initial scene or, as it is actually done, they can be streamed to update the scene in real-time.

3.2. Offline authoring

The offline authoring process is executed before the start of the sport event. It consists of preparing and encoding the initial presentation for each target terminal. The initial presentation, sent to the end-users' terminals at the start of the event, contains all static content, i.e., the initial BIFS scene, Object Descriptors (OD), and Elementary Streams (ES). For convenience, in our framework we decided to put all ESs (typically images) inside the initial scene. Indeed, all static content – such as images depicting the players or the sponsors in a football game – are most likely to be edited before the event. All the possible visual enhancements to display during the event are likely to be designed in advance as well. Therefore, we also include all possible visual enhancements in the initial scene. These graphics are given default parameters and are not necessarily fully specified, nor displayed in the initial scene. However, they stand as anchors in the scene for their future update during the online authoring process.

In our system, we prepare one initial presentation for each target terminal, i.e., PDA and STB, since they have very different characteristics in terms of display or inputs (e.g., the STB uses a remote control to interact with the scene, while the PDA uses a pen). Additionally, this approach allows fine adaptation of the look-and-feel of the presentation to the display for a particular terminal, thereby maximizing the user experience.

The director is given several templates for initial presentations, written in XMT. For illustration purposes, we take the example of a football match between two football teams. The director is given two XMT template files, describing the initial presentation for STB and PDA. He edits the files to adapt to this match, for instance changing the text giving the name of the teams or the name of the players. The director also prepares new media for this event, typically images such as sponsor images, and includes references to these media in the initial scene. The editing of XMT files is done through a GUI that hides the intricacies of the XMT structure from the director (so he does not have to know XMT in order to be able to edit the templates).

Fig. 5 shows the architecture of the module used to edit the XMT templates. The template edition tool takes the XMT template, as well as a personalization file associated to each template as inputs. The latter describes, in XML, the elements of the template that can be modified by the director. The edition tool parses both the XMT template and the personalization file, and displays through the GUI the elements that the director is allowed to change in the scene (such as the text giving the name of the teams or the reference to the images used). The edition tool outputs an edited XMT file that is ready to encode into MPEG-4.

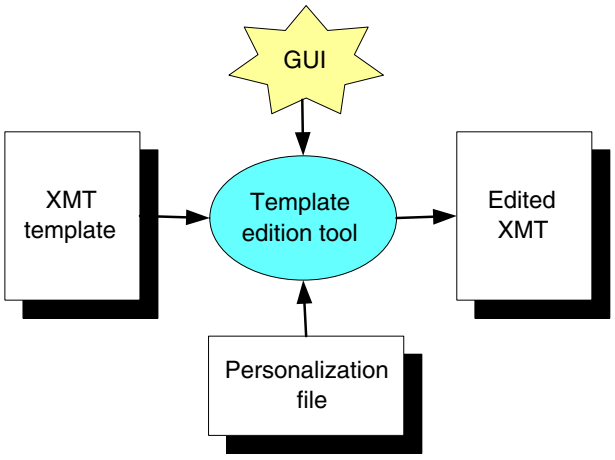

Fig. 5. Module for editing the XMT template.

Fig. 6 shows an example of a personalization file. This XML sequence is used to personalize the XMT template for a football game. In this example, the personalization sequence contains only one element to personalize, which is the name of football team number 1. Using XPath [8], the element points to the string attribute of the Text node, used to display the name of the football team in the template. A default value (here "ATeam") and a label attribute (here "name of football team 1") are assigned for easy retrieval by the director.

In our design, each XMT template is also associated with metadata describing the BIFS nodes in the initial scene that correspond to possible visual enhancements. For a given initial scene (see example in Fig. 8), the Authoring Engine needs to have a reference to the scene nodes for which it will trigger the generation of real-time BIFS Updates. Note that, in principle, the director can add new visual enhancements in the template scene, as long as corresponding metadata are added as well. The metadata are described in an XML file. Each BIFS node corresponding to a real-time visual enhancement is associated with what we call a *Customization Point*. Fig. 7 shows an example of Customization Point. This consists of three fields:

- an ID, which is the node unique binary ID within the initial presentation (we recall that nodes – or their field – can be updated only if they have been given an ID).
- a type, which is the node type (e.g. *Transform2D*, *Switch* ...).
- a human readable description, describing the graphics element this node corresponds to (This will allow the program director to select a real-time visual enhancement from its description).

After edition, the director sends the modified XMT template, and the offline media to use for the event (images) to

```
<perso target_url="football_stb.xmt">

<element path="/XMT-A/Body/Replace/Scene/OrderedGroup/
children/Switch[@DEF='N24']/choice/Transform2D/children/Shape/geometry/Text[@string]"
label="name of football team 1" type="string" default="ATeam" />

</perso>
```

Fig. 6. Example of personalization file.

```
<custoPoint id="17" type="IndexedLineSet2D" description="coordinates of line for distance to goal visual enhancement" />
```

Fig. 7. Example of customization point.

Fig. 8. Initial scene for football game on a STB.

Fig. 9. Example of real-time visual enhancement.

the offline encoding module. The module encodes the scene into an MPEG-4 file containing the initial presentation together with all offline media. At the start of the event, this MPEG-4 file is streamed by the Streaming Server to the end-users with the corresponding target terminal (see Fig. 8).

3.3. Online authoring

The online authoring process is in charge of the generation, encoding, and transmission of BIFS Updates that will update and/or display real-time visual enhancements during the event. BIFS Updates are generated from high-level interaction and graphics commands, which are selected by the program director during the event. To apply an Update to the right BIFS node, the director will select the corresponding Customization Point from its description (present in the metadata). Finally, the encoded BIFS Updates are sent to the Streaming Server for streaming to the end-users. Since all possible visual enhancements are placed in the initial scene, online authoring actually consists of updating and/or activating these enhancements. In particular, this permits streaming only small size binary information in real-time, instead of sending the replacement for the whole initial scene each time the director wishes to display or update a visual enhancement.

Fig. 9 shows the display of a line from the ball to the goal keeper and the corresponding distance on top of the image. The real-time display of the visual enhancement for the line results from the following high-level commands [1]:

- **changeLine** applied to a given *IndexedLineSet2D* BIFS node from the initial scene, with parameters giving the positions of the ball and the goal keeper (this position is obtained automatically from the director's authoring GUI as explained in Section 2).
- **showObject** applied to a given *Switch* node from the initial scene.

4. Digital Items in the proposed framework and implementation of the MPEG-21 standard

In the MPEG-21 [9] framework, complex digital objects are declared using the Digital Item Declaration Language (DIDL). This language defines the relevant data model via a set of abstract concepts, which form the basis for an XML schema that provides broad flexibility and extensibility for the actual representation of compliant data streams.

The usage of the MPEG-21 Digital Item Declaration Language to represent complex digital objects, has introduced benefits to the system in two major areas: The management of the initial presentation templates and the

management and distribution of multimedia content, such as video, images, and metadata. The platform allows the creation of pre-defined templates during the planning process before broadcasting; these templates form the Initial Scene that is used to generate the initial MPEG-4 scene. During the broadcasting phase, templates are used in order to control the real-time updates of the MPEG-4 content. Having all the information packaged in one entity, i.e., initial scene, customization points, etc. brings the benefit of reduced complexity data management.

Furthermore, the benefit from the adoption of MPEG-21 is that every Digital Item can contain a specific version of the content for each supported platform. The dynamic association between entities reduces any ambiguity over the target platform and the content. Having all the necessary information packaged in one entity enables the compilation and subsequent adaptation of a Digital Item to be performed only once (during its creation) and not on a per-usage basis, thereby effectively eliminating the need for storage redundancy and bringing the benefit of reduced management and performance complexity in the Information Repository. The adopted MPEG-21 concepts are described in the subsequent sections.

4.1. Multimedia resource adaptation

The focus of resource adaptation is the framework of DIA (Digital Item Adaptation), where messages between servers and users are in the form of XML documents with URL links to resources or encoded binary data. In the case of linked resources, a Digital Resource Provider decides which variation of the resource is best suited for the particular user, based on the user's terminal capabilities, the environment in which the user is operating and the available resource variations. For example, if the user wants to view a streaming media resource, adaptation will depend on the available bandwidth, screen size, audio capabilities, and available viewer software in the terminal, all part of an automated process, as capabilities and preferences should be automatically extracted and enforced.

In the described platform dynamic media resource adaptation and network capability negotiation, is especially important for the mobile paradigm (PDA's case) where users can change their environment (i.e., locations, devices. etc.) dynamically. MPEG-21 addresses the specific requirement by providing the Digital Item Adaptation (DIA) framework, for scalable video streaming [10].

The DIA framework, which is already discussed in the paper, specifies the necessary mechanisms related to the usage environment and to media resource adaptability. This approach was adopted for the MELISA platform. Alternative approaches to this issue may be the HTTP and RTSP based [11–13] or the agent based content negotiation mechanisms [14].

The DIA framework, regarding resource adaptation, includes the Usage Environment Description Tools (i.e., User Characteristics, Terminal Capabilities, Network Characteristics, and Natural Environment Characteristics) and the Digital Item Resource Adaptation Tools (i.e., Bitstream Syntax Description – BSD, Adaptation QoS, Metadata Adaptability). The BSD language and the Adaptation QoS schema are the two main tools, which enable resource adaptation. BSD language provides information about the high-level structure of bitstreams so that streaming can be modified according to this information. Adaptation QoS schema provides the relationship between QoS parameters (e.g., the current network interface bandwidth or the PDA's computational power) and the necessary adaptation operations to be executed for satisfying these parameters. The associated video or media quality, which is the outcome of the adaptation procedure, is also included as parameter in the adaptation schema. BSD is used in conjunction with the Adaptation schema to enable Quality of Service features. The adaptation procedure is as follows:

A normal DI and a DI with metadata are the inputs in the Digital Item Adaptation Engine. The engine operates on the input DIs according to the DIA tools. The engine is divided into two sub-engines: Description Adaptation Engine and Resource Adaptation Engine. The Resource Adaptation Engine performs adaptation on the corresponding resources and the result is the adapted DI which may contain the normal DI's contents, as well as the adapted DIA elements, Resources, DID Identifiers and IPMP/REL [15] elements.

When the client receives the DID, the user is prompted to make selections on Media Quality, while other parameters such as the presentation templates, the format and language would be selected automatically according to the terminal descriptors of the consumer device. These usage environment descriptors are incorporated into the middleware component. When the MELISA server receives the selection, it sends a positive or negative acknowledgement to the client side, according to the capability of the requested media resource to be adapted. For implementing dynamic resource update, the MELISA system allows the user to change the Media Quality choice during the video streaming. This approach can be easily extended to support network feedback, QoS decision module and automated video quality change according to current network status.

4.2. Content authoring and management

The system includes a range of authoring tools for production, encoding, and playback of interactive multimedia content in MPEG-4 for a variety of devices over fixed, wireless, and digital television networks [16]. The Multimedia Production Tools incorporate MPEG-4 and MPEG-7 content creation modules for encoding transition over DVB.

The platform foresees the infrastructure to support intelligent real-time game statistics and enhancements, utilizing information from various sources, both historical and during the events. This approach aims at providing

the viewer with valuable information presented in natural way, anywhere, thus increasing their intent in sports broadcasts. Advertising authoring tools aid the production and placement of dynamic advertisement of sports-related and other products. The system allows dynamic scene generation based on predefined templates. The use of these templates allows broadcasters to prepare their visually enhanced and interactive broadcasts well in advance, thus providing this service even during live events. Fig. 10 shows an overview of the server-side architecture, adapted to include the necessary concepts for rights management. DIs serve as the system input and stored in the main repository as links to binary files. During this authoring process, the relevant principles are recognized for the specific resources that they contribute (MPEG-2 video broadcasts, advertisements, visual enhancements, tracking information, and betting options). Information on the rights of the different users, based on the available subscription levels are encoded in the form of usage conditions and included in the resulting BIFS file, provided as the output of the multiplexing procedure and transmitted to the end-user terminals.

4.3. Initial presentation templates

A template is defined as a collection of screen elements that are subject to a task order. The task order implies that the screen elements are dependent on the completion of specific tasks. A screen element can vary from a simple logo, to placeholders on the screen where a textual description or even an image can be displayed. Regardless its shape, the screen element is considered to cover a rectangle box. The Predefined Templates in our system are used in two ways. Each template dictates the actual look of the broadcasted video. The menus, the panels, and the images that are used to display information and navigation elements to the viewer are all specified in the templates. Furthermore, for every broadcast, the templates are subdivided into one for the Set Top Box and one for the PDA client. The templates are made of static images and an XMT-A scene and stored in the Repository using a series of tables to store information for each template and template element.

This design is very flexible so that a template can contain any number and type of elements. A complex database structure allows the definition of a screen element as an object of any type, e.g., text, image, and video. It also allows template grouping so that a top level template would contain any number of templates that form the initial scene.

The disadvantage of this design is that the original XMT-A initial scene is lost, since the actual displayed content is revised via the transmission of updates and thus only the database representation is used at run-time. A large number of elements in the initial scene have a scene specific ID called a customization point, e.g., an on-screen button that displays the Lottery Options, or a statistics information area (see Section 3.3). Having the initial template divided into smaller templates and stored in the database calls for a complex customization point management. Essentially, the Information Merging Unit has to dynamically go through a group of templates and create a list of all customization points, so that in run-time, the Information Merging Unit (IMU) would update the correct customization point with the correct data. The IMU needs to handle

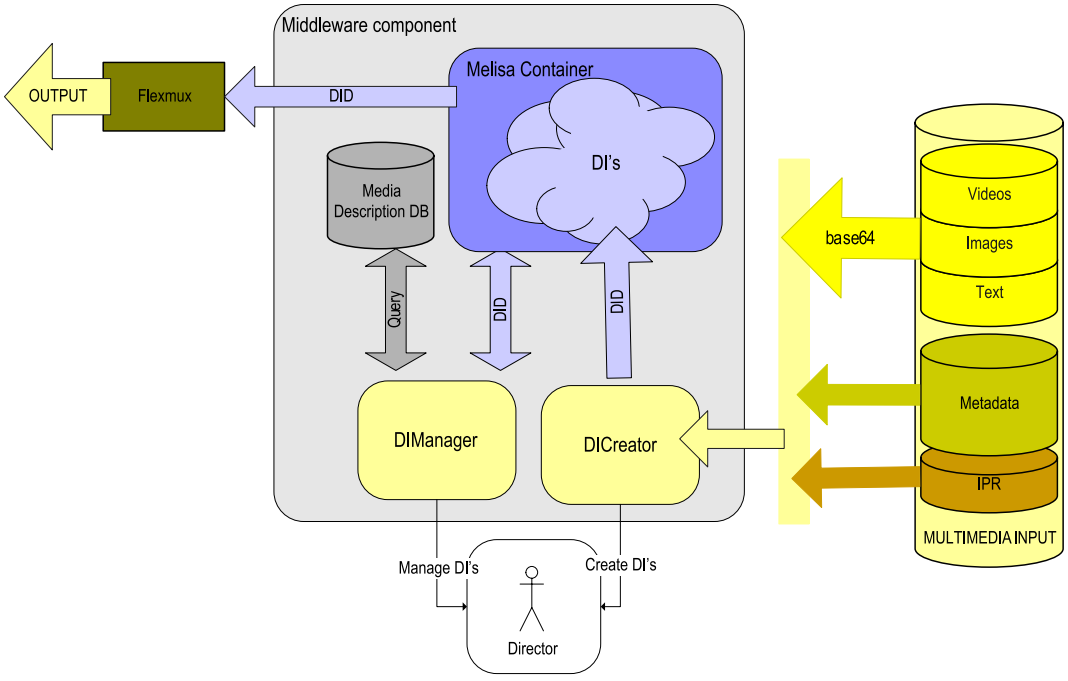

Fig. 10. The Server-Side Architecture with an initial distribution of roles in the most probable Digital Item model.

a number of templates at a time since each delivery platform would require instances of different templates.

The IMU generates the scene updates by modifying the properties of customization points in the scene. Combining different templates requires a customization point mapping and management module. The inclusion of all information within a DI implies that the customization points for every template are already specified. Embedding the customization points within the DID allows to associate the list of points to a particular content, while it remains readable by a human operator. In addition to this, representing the content as a DI may allow to add expressions for rights and protection within the DID. The DID framework also permits the use of MPEG-7 metadata as well, in order to describe semantics of the content.

The template DIDs use a very simple four-level hierarchy: Container/Item/Item/Component. This hierarchy allows the DIDs grow in width, rather than in depth through the addition of Items to a Container, Components to an Item, and Resources to a Component. Each DID container contains the DI id so that the Digital items can be clearly identified. Each container has an item which in effect contains the two templates as components, the Set Top Box template and the PDA Template. The component contains a resource that could be text, an image, or a video clip. The binary types of resources are basically URL references to the actual resource.

Obviously, with the Digital Item approach, accessing and modifying single elements in a template is not straight forward, since the template as a whole would have to be loaded in the authoring tool. The previous design with template grouping made the whole concept more efficient. Nevertheless the database design becomes simpler which has an effect not only during the offline editing but also during the broadcasting phase.

4.4. Video content authoring

The DI concept as proposed by MPEG-21 is utilized for the downloaded clips or images that are transmitted from the Server to the Set Top Box or a mobile device. The creation and usage of the DI can be divided into two parts. The first part is the DI creation that requires the collection of information from automated and user input and the second part is the DI management and streaming to the client platforms (Fig. 1). This methodology is common whether we are dealing with adverting clips or enhanced content replay clips. In the first case, the DIs are generated in a pre-broadcasting phase. The advertising unit receives the video clips and the DI items are generated and stored in the Information Repository. In the latter case the same method is followed, only that this time the process is performed during the broadcasting phase.

The Content DIDs use a 3-level hierarchy: Container/Item/Component. The Container can contain a number of Items that could be the video content for each supported platform, as seen in the Template Digital Items. Each Item contains a number of Components. In the example given below, we have an Item with three components, one containing the metadata description, one for the actual video content, and one that contains an image. This could be a Digital Item containing an advertising clip. The MPEG-7 metadata will be used at the client platform to decide whether to display the content, and if that is the case, the image containing an advertising message would be displayed prompting the user to view the advertising clip.

4.5. Final content representation

One of the goals of the described system is the interoperability of the interactive sport service over STBs and PDAs. Interoperability across different terminals is a common goal with the MPEG-21 framework. In our approach, the author of the rich media content prepares two versions of the initial presentation, one for the PDA and one for the STB. This is dictated by the fact that both terminals have very different display characteristics and that in professional applications the author of the content needs to create the best possible content for a particular display device.

The rich media initial presentation for a particular terminal is associated with a list of customization points which describe the BIFS nodes of the presentation that can be acted upon during the event (i.e., for which Updates will be generated). Both versions, PDA and STB, have a separate scene tree, but can have the same customization points. The initial presentations for both versions and the list of customization points must be known before the start of the event. For a given event, these data can be packaged in an MPEG-21 Digital Item, the DID will contain the list of customization points, as well as pointers to the alternative resources i.e., one .mp4 file for the STB and one for the PDA.

4.6. Personalization and content filtering

The system provides end users with the possibility to see only information that they are interested in. One flexible way to perform content personalization (bet menu, statistics, user-specific commercials, etc.) is to filter the BIFS Updates that are streamed to the client. In the case of display on STB, since the same BIFS Updates are broadcast to all clients, filtering should occur at the client side, i.e., on the STB before display.

The MPEG-21 framework is used for personalization and content filtering in the following way. The STB of a registered user contains an MPEG-21 DIA Description that specifies the user preferences on content. When the user terminal receives a BIFS Update, this is filtered according to its genre and the user preferences indicated in the DIA Description. The main issue is to find a way to synchronously transport the BIFS Update and its associated metadata indicating its genre, in order to make sure that the Update is not received before its description. One way to achieve this is by grouping the BIFS Update and its

genre within a DID, and to stream the complete DID to the clients. Below is one example of such a DID. This indicates that the Update belongs to the genre "Statistics". Obviously, according to the user preferences in the previous DIA Description, in this case the Update will be filtered out by the client terminal and therefore not displayed.

The main element (root element) extending the MPEG7-Type definition is shown below (Fig. 11). The original MPEG-7 DescriptionUnit definition has been restricted to include as child elements either the ViewerProfile (and optionally when applicable the ViewersProfile) element or the Sport Event (and optionally when applicable the MPEG-4 Content) element. Thus, it can generate instance root elements either for use of profile generated XML files at the client side, or metadata produced XML files at the sender production side. The ViewersProfile element groups together a number of ViewerProfile elements by referencing to them, thus producing a group of Viewers definition. The ViewerProfile element defines a viewer profile, his identification along with his MELISA specific preferences. The SportEvent element is the main element that contains all semantic information about a major sport event and sub-events, participating teams/athletes, phases of events (and involved athletes, teams), the broadcaster(s) covering the event, the sport itself, the sponsors of it, etc. as shown in the respective figure of the SportEventType definition (extending the MPEG-7 DSType). It is noted here that the SportEvent element is defined in such a way to serve both for the instantiation of information regarding major Sport events, as well as the instantiation of information regarding a specific sub-event in the major sport event. This is why the event phase information is used in a choice case to be instantiated only in the latter case, whereas in the former the necessity for including information of specific sub-events exists.

The MPEG-4 Content element (Fig. 13) describes all the information that is included in the respective MPEG-4 data produced. This case corresponds to non-live metadata transmission, metadata downloading, due to technology limitations. Some of the information may be appearing in *SportEvent* (Fig. 12) child element/attribute instances so the correspondence is maintained by referencing. The elements contain information such as the enhancements produced, the lottery event, the advertising content, functional information such as the interaction node definitions to drive the client applications, the spatial dimensions of the produced content, the color used as transparency if the Transparency node is not supported (see Fig. 13).

The *preferences* elements (Fig. 14) defined for the MELISA purposes are shown below. These involve preference registration on the sport event or location, favorite broadcasting channel (creation preferences), favorite sports, and preference on viewing related sport description, the types of visual enhancements he prefers to view, the types of bets (if any) he likes to place, his favorite athlete/teams/statistics, etc., or the language he likes all information to be encoded into. The preference value attribute that accompanies every different preference definition denotes the significance of that definition in the final calculation of the filtering decision metric.

4.7. Digital item generation

Two basic APIs were implemented for the creation and management of DIs, the MELISA Digital Item Creator (MDIC) API and the MELISA Digital Item Manager API (MDIM) (Fig. 15). The MDIC utilizes information both from the content author, as well as from automated processes at the sender side. Initially, the content author includes all the necessary information that the DI will contain, i.e., the video content. In this stage, the IPR information could be added in the DI, as specified using REL. The system was designed to generate metadata information to describe the content based on MPEG-7. The information describing the video clip is collected from the Information Repository, encoded in XML form and is included in the DI. The DI with the collected information and the multimedia content (video clip, image, etc.) encoded in base64 is then passed to the MDIM.

The MDIM separates the encoded visual information from the XML representation and maintains only a URL reference to it; this is performed to minimize data redundancy and speed up system response. The DI information is also stored in the Information Repository for later use.

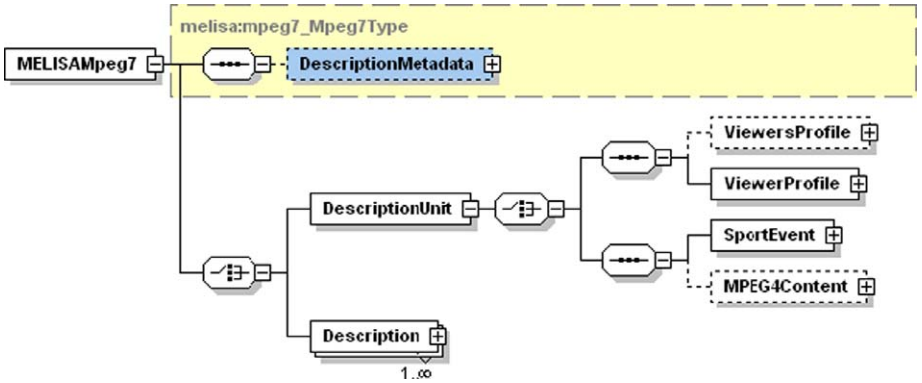

Fig. 11. MPEG7 Type definition.

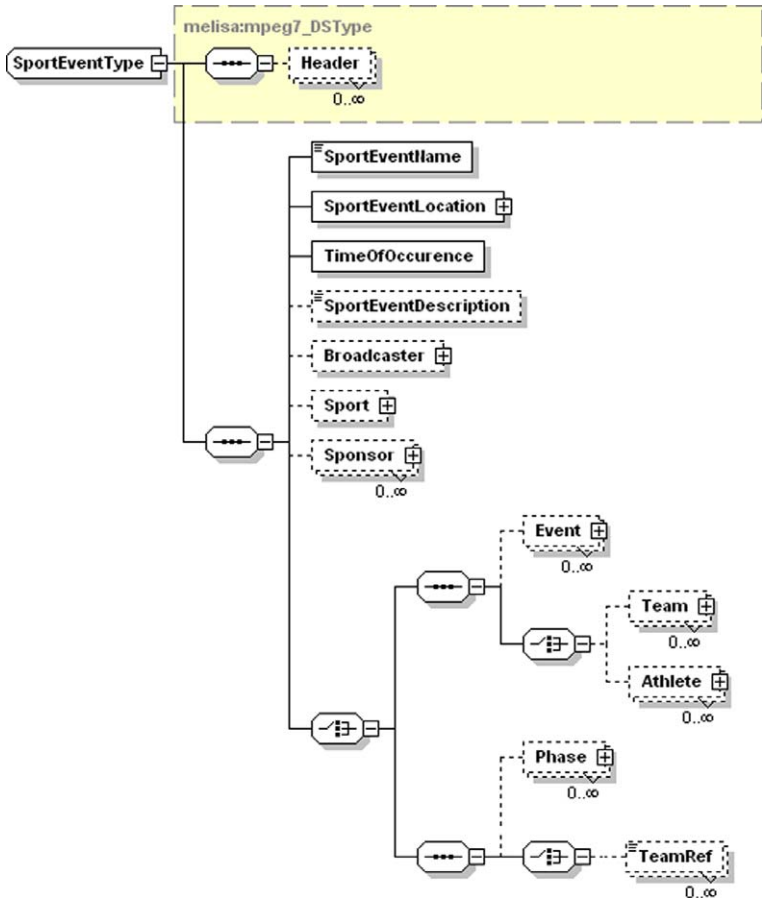

Fig. 12. Sport Event Type element.

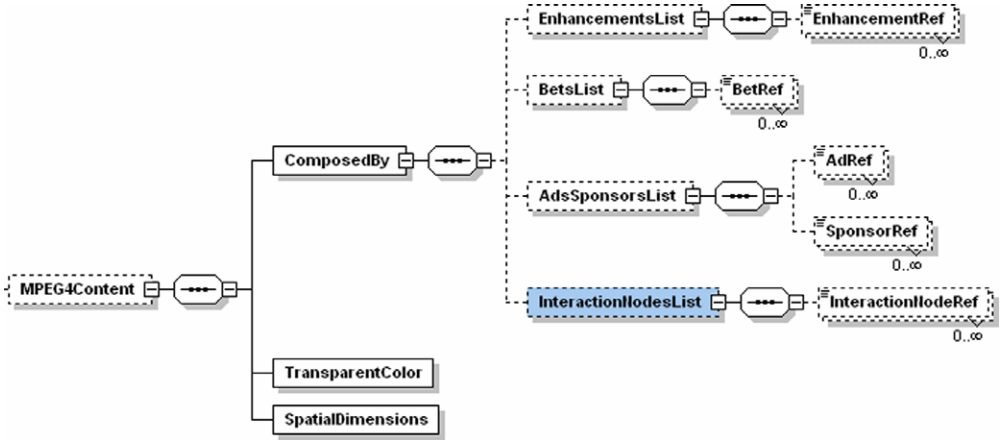

Fig. 13. MPEG-4 content element.

Should the director decide to transmit a DI, this is retrieved and then serialized in the form of a Java object. This serialized Java Object is then transformed into a binary array with a description header, ready for FlexMux [17] encoding and transmission.

The encoded and transmitted information is received by the client platform, where it is converted back to its original form and then received by the local DIM. User adaptation or Profiling is performed at this stage since all the necessary filtering information is included in the MPEG-7 metadata. The DIM filters the resource and decides whether to reject or accept it. The obvious advantage of having all the metadata in the DI and not in a separate stream is that at the Client-side timing issues between the download clips and metadata do no longer exist.

The basis of the MDIC API is the actual DIDL (Digital Item Declaration Language) XML elements. Every single

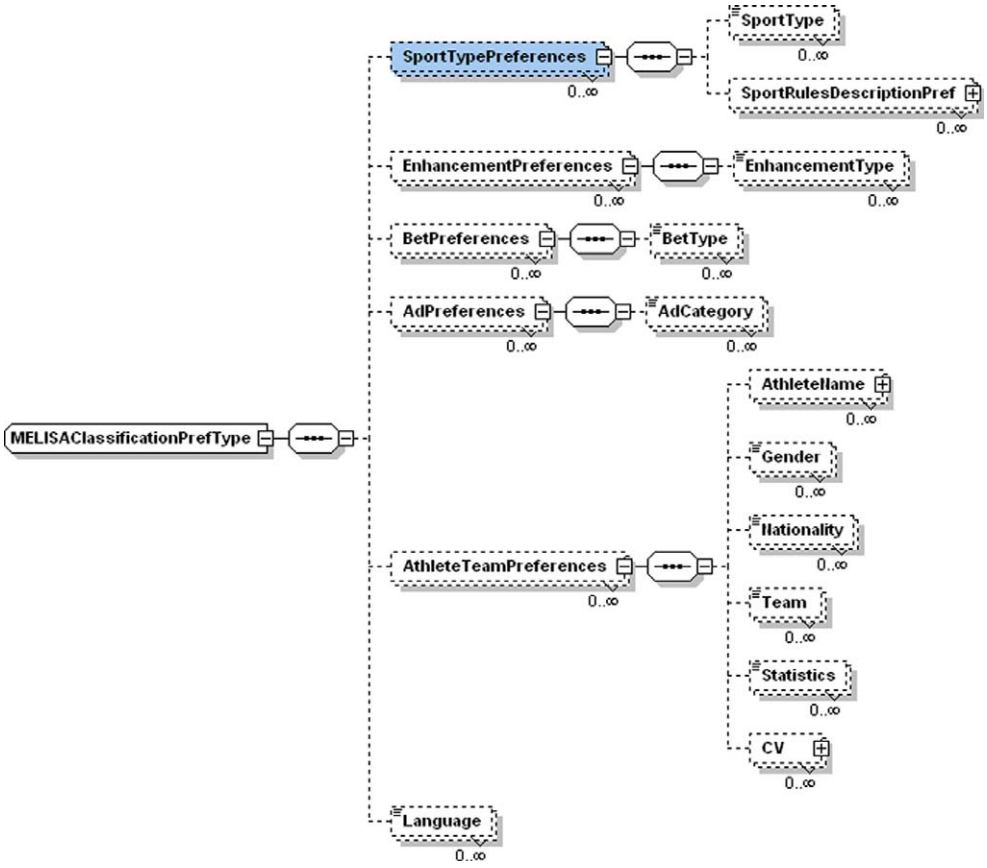

Fig. 14. Preferences elements.

MDIC class has a corresponding XML element in the schema, as generated using the Castor Java XML Data Binder. We chose the data binding method so that we can easily read and write valid XML documents. This adds flexibility to our system since any changes to the DIDL schema can be easily incorporated to our system, by simply re-creating the Java classes. The initial system was designed to transmit metadata in the form of serialized Java Objects, so this approach is maintained in the new design. The necessity of working and transmitting Java Objects was dictated by the Set Top Box limited processing power. Rather than transmitting XML data and having to parse them to extract the information, the information is already in the Java Objects.

The MDIC architecture can be conceptually divided into three layers (Fig. 16). The lower layer contains all the classes that where generated from the XML DIDL schema. The middle layer, the Creator class, contains a group of methods that enable the creation of any element of the DIDL schema. The creator class is responsible for updating the third level element object with the required information and returning the object to the top layer. The top layer, basically the MDIC API, is responsible for the creation of the DIs. This layer requests the element objects from the Creator class in order to form the Digital Items. The generated Digital Item are then either stored in the Information Repository, or exported to XML format files. Additionally a Digital Item can be provided to the system in XML form, to be imported to the system.

The MDIC is integrated in the whole information flow as an agent responsible for handling the offline and online processing of the Digital Items and is connected to the work flow control in order to be able to accept notifications concerning the available content. The Work Flow Control receives a notification from the Visual enhancements Unit every time new video content is generated. The Workflow control retrieves the Video Clip and notifies the MDIC that a new clip is available. The MDIC operator retrieves the basic metadata information in order to form the Digital Item, i.e., event information, description DI IDs. Metadata information is requested from the Metadata Manager and optionally from the MELISA Digital Rights Manager (MDRM). The information is then passed to the MDIM where it is stored in the Information Repository. The MDIM notifies the Work Flow Control that a new DI is ready. The director can see the list of the available DIs, and decide whether he will broadcast it on not.

4.8. Client-side application

The receiver platforms supported by the system are high-end set top boxes, portable digital assistants, and Java MIDP-enable mobile phones. The system provides content adaptation according to terminal capabilities, e.g., adapted

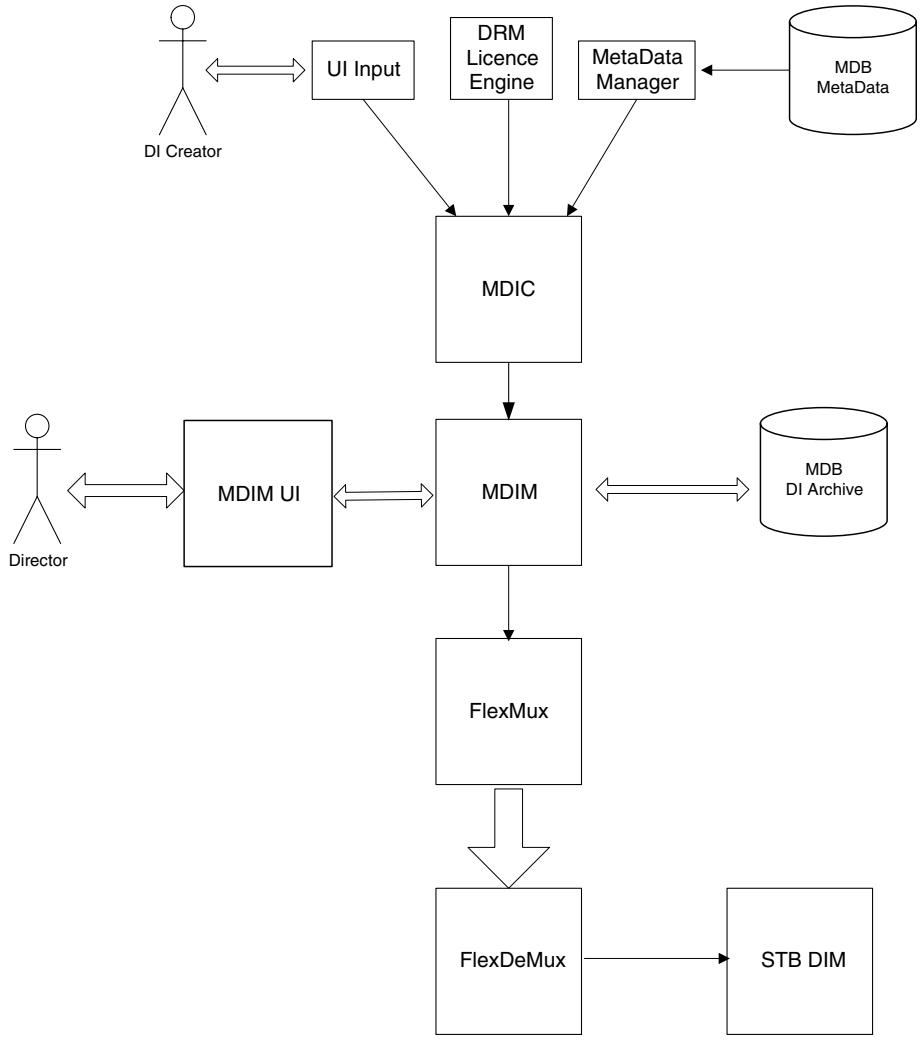

Fig. 15. Logical Diagram and data flow during the authoring, encoding and reception processes.

interaction, visual presentation in high or reduced resolution graphics, etc.

As shown in Fig. 17, the receiver initially decodes the incoming encoded streams to gain access to the stored visual information and the associated metadata. As stated earlier, the Digital Items consist of visual information in the forms of MPEG-2 video and MPEG-4 graphics, multiplexed with MPEG-7 metadata and MPEG-21 information into an MPEG-2 stream to be transmitted.

Each broadcasted event can be conceptually divided into segments, according to the program planning performed by the broadcaster. Every event is planned in advance and divided into segments, e.g., the first half of a football game, the 15-min advertising break during half time, the first section of a Grand Prix, the 5-min advertising breaks during the race, etc. Each segment can be described using a Digital Item that basically contains the MPEG-7 metadata description.

The stream is received by the Set Top Box and the FlexMux stream is decoded by the FlexDemux module. The DI

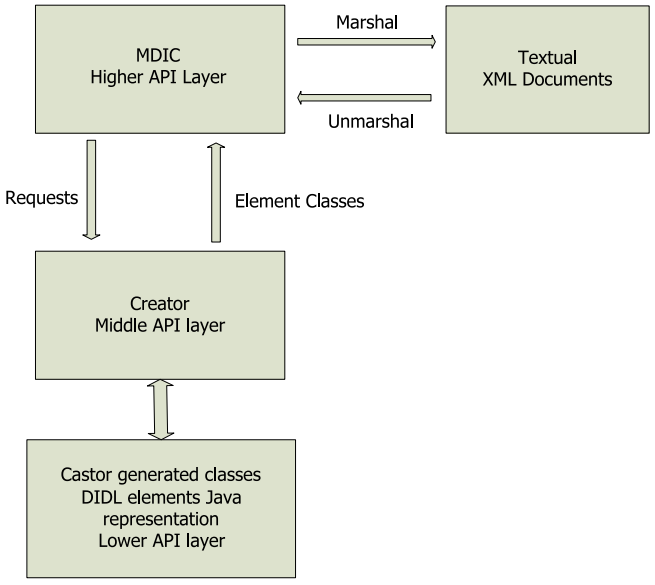

Fig. 16. MDIC architecture diagram.

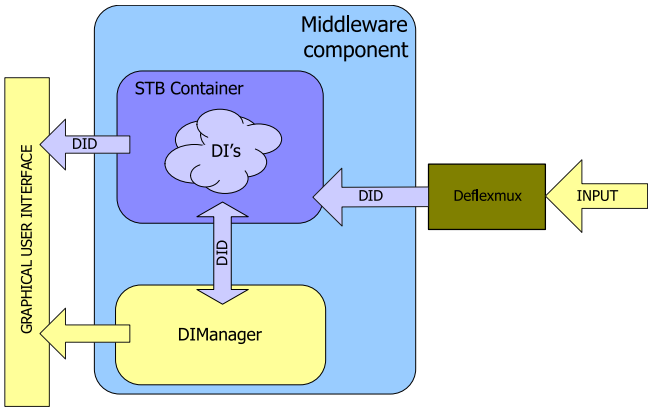

Fig. 17. The Client-Side Architecture with an initial distribution of roles.

and the incorporated MPEG-7 metadata contain the information necessary to describe the resource and the related rights. Since this is a multicast environment, rights enforcement and profiling has to take place in the receiver, after the decoding process. Essentially, if the user (principal) logged on to a particular terminal has the rights necessary to perform specific actions on the included content, and his personalized profile allows it, the STB container will forward the binary and text data to the user interface for playback.

In the case of offline video replays (downloaded clips), these are now streamed as Digital Items. As the Digital Items arrive at the Set Top Box are de-multiplexed, and the local MDIC takes over to handle the Digital Items. The local Profile Manager is consulted to make a decision on whether the user would like to view this type of content. If the local profile allows the playback of such content, it is displayed to the user, otherwise it is ignored.

5. Conclusion

The proposed system was originally designed to handle video content in a conventional way. The investigation of MPEG-21 showed that the adoption of such a standard brings a number of advantages. Packaging the enhanced content into MPEG-21 Digital Items allows us to include time critical data, such as MPEG-7 metadata descriptions, consequently eliminating certain synchronization issues during broadcasting, while the concept of Digital Items allows us to include information necessary to update the MPEG-4 scenes in real-time. Having to maintain a mapping of the relevant data between the initial MPEG-4 scene and the actual stored templates was a significant burden; having everything predefined and included in the Digital Items of the initial scene simplifies the template management.

Another significant benefit from the adoption of MPEG-21 is that every Digital Item can contain a version of the content for every supported platform. The dynamic association between entities reduces any ambiguity over the target platform and the content. Having all the information packaged in one entity brings the benefit of reduced complexity in the Information Repository Design, although at the same time somewhat reduces overall flexibility. Nevertheless it is believed that the adoption of MPEG-21 would bring significant benefits in the MELISA Platform, especially in the case of protecting the intellectual property of the different vendors (TV stations that do the capturing and processing, betting and sports statistics companies, advertisement companies, etc.) that contribute in the distributed content. The Digital Rights Management infrastructure that the MPEG-21 standard proposes is possibly its most significant aspect when it comes to multi-vendor content, such as the one prepared and delivered in the MELISA system. In this framework, the adoption of Digital Items, instead of merely packaging the audiovisual material in MPEG-4 scenes, enables the final content provider to provide different subscription levels, each one with different rights, thus honoring the contribution of each content vendor in accordance to the usage of the relevant content. Indeed, this seems to be one of the most valuable benefits of the adoption of MPEG-21 concepts within the MELISA system, enabling content integrators to focus on the actual services that they provide, without having to worry about any technological issues involved in this process.

Future work regarding the integration of diverse services lies in the field of introducing MPEG-7 related schemas to provide group-based content filtering and transparent utilization of other information resources (video footage archives or statistics sites) to provide semantically related material to the users (e.g., highlights from previous matches between two particular teams); the provision of play-by-play statistics, along with MPEG-7 technology, enables the receiver to provide the end user with alerts related to specific events or keep statistics based on the events that are of interest to the user, e.g., correct/incorrect passes during a football match. Again, DRM technology allows contributing third parties to retain the rights of their material, while still receiving fees with respect to actual usage.

Acknowledgments

MELISA is an EU-funded 5th framework IST project (IST-2001-34755). The authors thank all partners of the project: Intracom S.A. (coordinator), Ogilvy Interactive S.A., Symah Vision, University of Essex, Intralot S.A., Cosmote S.A., Uppsala University, Ladbrokes Ltd., ERT – Hellenic Broadcasting Corporation, ENST – Ecole Nationale Supérieure des Télécommunications and Ondim. For more information, visit http://melisa.intranet.gr.

References

[1] E. Papaioannou, K. Karpouzis, P. de Cuetos, V. Karagianis, H. Guillemot, A. Demiris, N. Ioannidis, MELISA – a distributed multimedia system for multi-platform interactive sports content broadcasting, in: Proceedings of EUROMICRO Conference, 2004, pp. 222–229.

[2] Coding of Audio-Visual Objects (MPEG-4) – Part 1: Systems, ISO/IEC, 2001, 14496-1.
[3] Computer Graphics and Image Processing (VRML) – Part 1: Functional Specification and UTF-8 Encoding, ISO/IEC, 1997, 14772-1.
[4] C. Concolato, J.-C. Dufourd, J.-C. Moissinac, Creating and encoding of cartoons using MPEG-4 BIFS: methods and results, IEEE Transactions on Circuits and Systems for Video Technology 13 (11) (2003) 1129–1135.
[5] C. Concolato, J.-C. Dufourd, Comparison of MPEG-4 BIFS and Some Other Multimedia Description Languages, in: Proceedings of WEMP Workshop, June 2002.
[6] J.-C. Dufourd, BIFS: Scene Description, in: F. Pereira, T. Ebrahimi (Eds.), The MPEG-4 Book, Chapter 4, IMSC Press, Prentice Hall, New Jersey, 2002.
[7] ISO/IEC JTC1/SC29/WG11/N3382 14496-1:2001 PDAM2 (MPEG-4 Systems), Singapore, March 2001.
[8] W3C XML Path Language (Xpath) Recommendation, November 1999. <http://www.w3.org/TR/path/>.
[9] MPEG-21 Overview v.5, ISO/IEC JTC1/SC29/WG11/N5231, Shanghai, October 2002. <http://www.chiariglione.org/mpeg/standards/mpeg-21/mpeg-21.htm/>.
[10] A. Perkis, Y. Abdeljaoued, C. Christopoulos, T. Ebrahimi, Universal multimedia access from wired and wireless systems, Birkhauser Boston transactions on circuits, systems and signal processing, Special Issue on Multimedia Communications 10 (3) (2001) 387–402.
[11] K. Holtman and A. Mutz, Transparent Content Negotiation in HTTP, IETF RFC 2295, 1998.
[12] Apache Group, Apache HTTP Server: Content Negotiation. <http://httpd.apache.org/docs/content-negotiation.html/>.
[13] S. Seshan, M. Stemm, R.H. Katz, Benefits of transparent content negotiation in HTTP, Global Internet Mini Conference, Globecom, 1998.
[14] K. Munchurl, L. Jeongyeon, K. Kyeongok, K. Jinwoong, Agent-based intelligent multimedia broadcasting within MPEG-21 multimedia framework, ETRI Journal 26 (2) (2004) 136–148.
[15] M. Ji, S.M. Shen, W. Zeng, T. Senoh, T. Ueno, T. Aoki, Y. Hiroshi, T. Kogure, MPEG-4 IPMP extension for interoperable protection of multimedia content, EURASIP Journal of Applied Signal Processing 14 (2004) 2201–2213.
[16] S. Maad, Universal access to multimodal ITV content: challenges and prospects, Proceedings of User Interfaces for All 2002, 195–208.
[17] MPEG-4 Overview v.21, ISO/IEC JTC1/SC29/WG11 N4668, March 2002, <http://www.chiariglione.org/mpeg/standards/MPEG-4/MPEG-4.htm/>.

Dr. Kostas Karpouzis graduated from the School of Electrical and Computer Engineering of the National Technical University of Athens in 1998 and received his Ph.D degree in 2001 from the same University. His current research interests lie in the areas of human computer interaction, image and video processing, sign language synthesis and virtual reality. Dr. Karpouzis has published more than seventy papers in international journals and proceedings of international conferences. He is a member of the technical committee of the International Conference on Image Processing (ICIP) and a reviewer in many international journals. Dr. Karpouzis is an associate researcher at the Institute of Communication and Computer Systems (ICCS), a core researcher of the Humaine FP6 Network of Excellence and holds an adjunct lecturer position at the University of Piraeus, teaching Medical Informatics and Image Processing. He is also a national representative in IFIP Working Groups 12.5 'Artificial Intelligence Applications' and 3.2 'Informatics and ICT in Higher Education'.

Ilias Maglogiannis received the Diploma in Electrical & Computer Engineering and a Ph.D. in Biomedical Engineering from the National Technical University of Athens (NTUA) Greece in 1996 and 2000 respectively. From 1996 until 2000 he worked as a Researcher in the Biomedical Engineering Laboratory in NTUA. Since February of 2001 he is a Lecturer in the Dept of Information and Communication Systems Engineering in University of the Aegean. He has been principal investigator in many European and National Research programs in Biomedical Engineering and Health Telematics. He has served on program committees of national and international conferences and he is a reviewer for several scientific journals. His scientific activities include biomedical engineering, image processing, computer vision and multimedia communications. He is the author of sixteen (16) journal papers and more than fourty (40) international conference papers in the above areas. Dr. Maglogiannis is a member of IEEE – Societies: Engineering in Medicine and Biology, Computer, Communications, SPIE – International Society for Optical Engineering, ACM, the Technical Chamber of Greece and the Hellenic Organization of Biomedical Engineering. Dr. Maglogiannis is also a national representative for Greece in the IFIP Working Groups 3.2 (Informatics and ICT in Higher Education) and 12.5 (Artificial Intelligence – Knowledge-Oriented Development of Applications.

Emmanuel Papaioannou holds a Degree (B.Sc.) in Computer Science, a Masters (M.Sc.) in 3D Computer Graphics all from the University of Teesside in the UK. He is currently registered as a Part Time Ph.D. student in the area of 3D Visualization for Medical Imaging Applications at the University of Teesside. He has worked in various contracts in the area of Computer Graphics and Virtual Reality. He has worked as a research assistant, and part time lecturer at the University of Teesside as well as a Technical Project Manager in the Computer Integrated Quality Assurance project (CRAFT BES2-5388) related to Virtual Reality and Engineering. Before joining INTRACOM, he has employed at Industrial Technologies S.A. as a software engineer in industrial machines integration with Plant Management Systems. He is currently employed as Technical Coordinator for the IST project Melisa. His research interests include Image Processing and Visualization, Virtual Reality, Robotics, Human Computer Interaction and Digital Content Management.

Dimitrios D. Vergados was born in Athens, Greece in 1973. He is a Lecturer in the University of the Aegean, Department of Information and Communication Systems Engineering. He received his B.Sc. in Physics from the University of Ioannina and his Ph.D. in Integrated Communication Networks from the National Technical University of Athens, Department of Electrical Engineering and Computer Science. His research interests are in the area of Communication Networks (Wireless Broadband Networks, Sensor – Ad-hoc Networks, WLANs, IP, MIP, SONET Networks), Neural Networks, GRID Technologies, and Computer Vision. He participated in several projects funded by EU and National Agencies and has several publications in journals, books and conference proceedings. Dimitrios D. Vergados is a member of the IEEE. He is also Guest Editor and Reviewer in several Journals and member of International Advisory Committees of International Conferences.

Dr. Angelos Rouskas was born in Athens, Greece, in 1968. He received the five-year Diploma in Electrical Engineering from the National Technical University of Athens (NTUA), the M.Sc. in Communications and Signal Processing from Imperial College, London, UK, and the Ph.D. in Electrical and Computer Engineering from NTUA. He is an assistant professor in the Department of Information and Communication Systems Engineering of the University of the Aegean (UoA), Greece, and Director of the Computer and Communication Systems Laboratory. Prior to joining UoA, Dr. Rouskas worked as a research associate at the Telecommunications Laboratory of NTUA, in the framework of several European and Greek funded research projects, and at the Network Performance Group of the Greek Cellular Operator CosmOTE S.A. His current research interests are in the areas of resource management of mobile communication networks, mobile networks security, and pricing and admission control in wireless and mobile networks and he has many publications in the above areas. He is the TPC co-chair of European Wireless 2006 conference and has served as a TPC member in several international conferences. Dr. Rouskas is a reviewer of several IEEE, ACM and other international journals and a member of IEEE and of the Technical Chamber of Greece.