International Journal of Software Engineering and Knowledge Engineering

EXPLOLOTING INTELLIGENT CONTENT VIA AXMEDIS/MPEG-21 FOR MODELLING AND DISTRIBUTING NEWS

Pierfrancesco Bellini

DISIT-DSI, Distributed Systems and Internet Technology Lab
Dipartimento di Sistemi e Informatica, Università degli Studi di Firenze, Firenze, Italy
http://www.disit.dsi.unifi.it/ , pbellini@dsi.unifi.it

Ivan Bruno

DISIT-DSI, Distributed Systems and Internet Technology Lab
Dipartimento di Sistemi e Informatica, Università degli Studi di Firenze, Firenze, Italy
http://www.disit.dsi.unifi.it/, nesi@dsi.unifi.it, paolo.nesi@unifi.it

Paolo Nesi

DISIT-DSI, Distributed Systems and Internet Technology Lab
Dipartimento di Sistemi e Informatica, Università degli Studi di Firenze, Firenze, Italy
http://www.disit.dsi.unifi.it/, ivanb@dsi.unifi.it

Received (Day Month Year), Revised (Day Month Year), Accepted (Day Month Year)

The content technology needs to attain forms with more intelligence, flexibility and complete features than those being currently on the market or proposed by standards. In this paper, an analysis of the state of the art about intelligent and complex content models is presented. The analysis allowed identifying a number of topics and features which models and formats should evolve according to. The work has been used to extend AXMEDIS content model and format which in turn is grounded on MPEG-21, SMIL, HTML, and other standards. The Extended AXMEDIS format presents a set of new features among them: semantic descriptors, extended annotations, intelligent behavioral and semantic computing capabilities. The newly obtained format has been compared against NewsML which is one of the most widespread formats for news production and distribution. The management of news has some peculiarities such as container, production tools and players, that may take advantages of the intelligent content features and applications. Moreover, news have to be massively processed for ingestion and repurposing, and present relevant requirements on right control. Also these features may be satisfied by AXMEDIS tools . To this end, a comparative analysis of processing and modeling NewsML with AXMEDIS tools and format has been performed and reported to verify the usage. In addition, AXMEDIS format can be profitably used for a range of innovative applications of intelligent content.

Keywords: cross media, intelligent content, MPEG-21, NewsML, repurposing, automated content management, DRM

1. Introduction

With the evolution of multimedia and cross media a large range of content formats has been proposed. They range from the single files, based on simple resources, or essences such as documents, videos, images, audio, etc., to integrated content models for multimedia/crossmedia, such as MPEG-21 [1], [2], MXF [3], NewsML [4], [5], SCORM/IMS [6], MPEG-4 [7], METS [8], and proprietary formats such as Adobe Flash, MS Silverlight, etc. Among them, MPEG-21 is focused on the standardization of the content description including digital rights management, DRM, aspects [9], [10]. MXF [3] has been designed as an exchange format and to address a number of problems of non-professional formats. MXF has been designed as an exchange format and to address a number of problems of non-professional formats. MXF has full time code and metadata support, and it is meant to be a platform-agnostic stable standard for future professional video and audio applications. Integrated content formats wrap different kinds of digital resources/files in a container/package with their related information (e.g., content metadata and descriptors, relationships among resources, etc.) and they make such resources ready for delivery (streaming and/or downloading), in plain (clear-text) and/or protected forms. NewsML of IPTC (International Press Tele-communication Council) [4] provides support for both referencing textual news, resource files, and paging them, while collecting metadata and descriptors, vocabularies, etc. (http://www.iptc.org). NewsML and SportML belong to a family of news formats that cannot but demonstrate their current inefficiency to cope with the complexity of nowadays news. SCORM [6] is a comprehensive standard for the organization and delivery of learning packages. METS [8] is an XML format defined for coding information for interchange, including descriptors, administrative, and structural aspects which are needed in digital libraries. METS is strongly related to MARC metadata model and team [11], while it is not suitable for media and entertainment.

The presentation layers and user interactions are formalized with formats such as: SMIL [12], HTML, MHP [13], MHEG [14], Laser [15], Java, SVG [16], MPEG-4 BIFS [7], etc. SMIL is similar to an HTML-like language designed to produce interactive presentations, and it may have links to other SMIL presentations and graphical elements to allow user interaction. SMIL [12] provides features like transitions, animations, etc., and it is one of the underlying technologies used by DVD and MMS for advanced interactivity. The MHP [13] enables the reception and execution of interactive, Java-based applications on a TV-set. Interactive MHP applications can be broadcast together with audio and video streams. The applications can be on information services, games, interactive voting, e-mail, SMS or shopping. MHEG [14] is a standard for the formalization of the presentation/interaction aspects for broadcasting. SVG [16] is an XML specification and file format for describing vector graphics, both static and animated. The W3C explicitly recommends SMIL as the standard for animation in SVG, however it is more widespread to find SVG animated with ECMAScript (JavaScript). BIFS is MPEG-4 Part 11 [7], it is a binary format for two and three-dimensional audiovisual content, with graphic rendering and interaction. BIFS is MPEG-4 scene description protocol to compose, describe interactions and animate MPEG-4 objects.

The metadata are frequently defined together with content ID codes. Among the metadata: Dublin Core [17], TVAnyTime [18], OAI ORE [19], MPEG-7 descriptors [20], MARC [11], etc. and among the identification codes, ISBN, ISAN, ISRC, ISMN, [21], etc., but also classical URI.

More recently, a certain number of formats has been presented with the aim of providing more advanced experiences to final users enforcing some degree of intelligence into the object format among them: ACEMEDIA [10], [23], X-MEDIA [24], [25], AXMEDIS [26], [27], [28], EMMO [29], and KCO [30]. ACEMEDIA defined a content format to enable creating personalized content collections. X-MEDIA content model is mainly focused on semantic aspects that can be managed by ontologies and RDF. X-Media is mainly oriented towards knowledge management and sharing with limited application to text and image contents and it has related content objects with very limited autonomy of work that are not proactive with the user. AXMEDIS is an extended version of MPEG-21 supporting DRM and proposing content packing with presentation capabilities in HTML, FLASH and SMIL [12], including behavioral capabilities and semantics descriptors. EMMOs (Enhanced Multimedia Meta Objects) encapsulates relationships among multimedia objects and maps them into a navigable structure. An EMMO contains media objects, semantic aspect, associations, conceptual graphs, functional aspect. KCO, Knowledge Content Objects, is not a package, and it is based on the DOLCE foundational ontology and has semantic aspects to describe the properties of KCOs, including raw content or media item, metadata and knowledge specific to the content object and knowledge about the topics of the content (its meaning). The semantic information in a KCO includes: content description; propositional description (Semantic Description and Content Classification); presentational description; community description (the purpose); business description (the trade, price…); trust and security description, self description (the structure).

Most of these last models present descriptors that may be used for powerful semantic classification. Some of them present also capabilities to formalize content behavior, for example in Java and/or JavaScript. Among the formats mentioned, the AXMEDIS implementation of the MPEG-21 file format and MXF [3] supports the direct play. Only the MPEG-21 also supports a range of business and transaction models via DRM (Digital Rights Management) solutions [9], [10] and with a set of technological protection supports.

This paper reports the developments performed in Extended AXMEDIS model and format as intelligent content and the modeling and distribution of news and news collections. The former AXMEDIS format (automated cross media content for multichannel distribution) has been defined as a media wrapper and packager, integrating and extending standards such as MPEG-21 [27], [31], [32]. AXMEDIS was an integrated research and development project funded by the European Commission. It has been developed by more than 40 partners including: University of Florence, HP, EUTELSAT, TISCALI, EPFL, FHGIGD, BBC, AFI, University Pompeo Fabra, University of Leeds, STRATEGICA, EXITECH, XIM, University of Reading, etc., and it ended in September 2008.

After that, the AXMEDIS framework has been strongly improved/extended with a large set of semantic tools and functionalities increasing model intelligence, back off automation capabilities, and producing smart tools for PDA, iPhone and a large set of browsers. The Extended AXMEDIS format includes full multilingual modeling for metadata and descriptors, semantic descriptors in RDF and XML, direct links and access to internal resources, extended behavioral capabilities, semantic computing capabilities, maintaining old packaging and DRM supports.

As to the paper organization, it is as follows. Section 2 reports an overview and comparison of the most widespread intelligent content models. This comparison aimed at putting in evidence the enabling technologies exploited by different models and their limitations. This analysis has been conceived to lay the foundations for the Extended AXMEDIS model presented in this paper, and it may be useful for the reader when it comes to identifying the best model for a given application. The Extended AXMEDIS model and format are supported by a set of tools for content authoring, execution and for massively scalable automated management of content, the so-called AXMEDIS content processing, AXCP, tools. AXCP is a grid architecture providing a set of technical solutions and tools to allow the automation of cross media content processing, production, packaging, protection and distribution.

In Section 3, the main aspects of Extended AXMEDIS Intelligent Content model and format are presented, considering: descriptors, behavioral, structural, presentation capabilities and tools: for execution and production. With the new solution and model, it is possible to exploit the AXMEDIS format and tools to produce and distribute intelligent content, and to exploit the legacy content formats such as: HTML, NewsML, MXF, SCORM, ZIP, SMIL, etc. Section 3.2 reports some examples of intelligent content, while complex and complete example can be recovered on the trial portals http://mobmed.axmedis.org and http://xmf.axmedis.org and by downloading the demo tools.

In Section 4, the usage of the Extended AXMEDIS format to cope with NewsML content is presented to highlight the most innovative aspects of the intelligent content and subsequently the advantages if compared with the traditional models. In the same section, there are also some comments about the automated management of NewsML with AXMEDIS tools. NewsML modeling and conversion strategy into AXMEDIS and MPEG-21 formats have been identified with the aim of preserving semantics and capabilities of the News files processed [4]. Section 4.1 describes the usage of AXCP tools for the automated ingestion, repurposing and distribution of news in several formats. AXCP tools are used for multimedia processing and can cope with a large number of formats including MPEG-21, XML, MFX, SMIL, HTML, XSLT, RDF, etc., and they can work with a multichannel architecture for the production of content on demand. Section 4.2 reports the advantages in using AXMEDIS for modeling cross media news. Conclusions are drawn in section 6.

2. Analysis of Intelligent Content Features

The main aim of intelligent content formats is to present a certain degree of autonomy and intelligence, which enables them to be active and proactive with users and not only interactive. These capabilities are grounded on a number of features to provide services to users. Some of them are available in current state of the art formats, whereas others are innovative aspects.
In Table 1, a taxonomy of the relevant features which allowed to assess the above mentioned formats is reported. The table reports only the most representative formats and models among the different categories mentioned before: standards, proprietary format, research projects, presentation models, etc. In the next paragraphs, an analysis of the most relevant factors and how they should be used to enforce intelligence content is reported. The analysis aims also at highlighting which are the main enabling technologies behind the formalization of intelligent content and tools. Most of the features described above are not only interesting for news. Similar problems of distributed content generation, annotation, self protection, knowledge modeling, etc., are also suitable for the User Generated Content, UGC.

Structural and file-format aspects have to deal with how an object is composed/arranged in terms of digital essences/files, metadata, presentation information, descriptors, annotations, other nested objects, and other pieces of information; and therefore, which are the relationships elements are arranged according to and how this information and data are packaged (i.e., references to external resources and/or resource embedding). The structural complexity impacts on content distribution and usage since multiple paths and non-linear links/stories are not simple to be streamed and accessed in real time when played. As to structural aspects, cross media formats are those that may contain other essences, for example AXMEDIS, SCORM, MXF, MPEG-21 DIDL [33], etc. In some cases, they can be also managed as groups of files glued by presentation models, such as HTML, SMIL. The structure has also to provide support for augmenting the content structure with additional content, metadata, descriptors, annotation, while integrating them into the package, which increases the complexity and may change the capabilities of the intelligent content. For example, the possibility of creating: personal collection of content, adding annotations and resources, creating new personal narrative paths in the complex content, creating interface elements dynamically and joining them to the content, migrating the content performing adaptation, etc.

The possibility of identifying the single cross media elements of complex packages with ***direct links*** allows the creation of annotations and direct access. Direct links allow direct access to the resources or to their segment (e.g., 12345 ms from the beginning, the single image of a sequence, the single tag of an HTML page). In many packages, the problem of defining relationships is addressed in terms of IDs and hyperlinks. For example, HTML and SMIL have links that can refer to digital resources located on the same place of the main file or located remotely with an URL.

	MPEG-4	MPEG–21	SCORM/IM	NewsML	MHEG	MHP	SMIL	HTML	MXF	BIFS	SVG	Laser	Flash	Silverlight	AXMEDIS	ACEMEDIA	X-MEDIA
Structural and file format																	
Cross media nesting levels	P	Y	Y	Y					P						Y	P	
Any kind of content files		Y	Y				P	A							Y	A	
Direct Links and navigation	P			P			Y	Y			Y	Y	Y	Y	Y	P	
Reflection on structural capabilities		Y												Y	Y		
Augmentation of content package		Y							Y				Y		Y		
XML and/or Text format	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y	Y			Y	Y	Y
Binary File format	Y	Y				Y			Y	Y	Y	Y	Y	Y	Y		
Packaging and wrapping	Y	Y	Y	A					Y		Y	Y	Y	Y	Y		
Descriptors																	
Simple Static Metadata	Y	Y	Y	Y	Y	Y	Y	Y	Y		Y	Y	Y	Y	Y	P	Y
Extendible Metadata		Y	Y	Y					Y		Y				Y	P	Y
Unique/Multiple identifiers		Y							P						Y		
Descriptors and semantics	A	A												P	Y	P	Y
IPR/DRM Descriptors		A		P		P									Y		
Reflection on content descriptors		Y													Y		
External Annotations	P	Y	Y										Y		Y	Y	Y
Multimedia annotation to internal details		P							P						Y		Y
Behavioral																	
Synchronization	Y				Y	Y	Y		Y	Y	Y	Y	Y	Y	P		
Internal functional parts	Y	Y	Y		Y	Y		A		Y	Y	Y	Y	Y	Y	P	
Communication capabilities		Y	Y	Y		A	Y	Y				Y	Y	Y	P		Y
Collaboration among users															P		Y
Reflection capabilities on behavior		P						P						P	Y		
Media Processing capabilities	Y	Y			Y	Y		A		Y			P	P	Y	P	
User profiling processing		Y				A							A	Y	Y		
Device and contextual profiling proc.	P	A			Y	A		P		Y		Y	A	Y	Y	P	
Taking decision engine		A				A							A		Y		
Presentation and Interaction																	
Integrated presentation layers	Y	A	A		Y	Y	Y	Y		Y	Y	Y	Y	Y	Y	P	
Animations	Y	A	A			Y	Y	A		Y	Y	Y	Y	Y	Y	P	
3D aspects	Y	A	A			P		A		Y	Y	Y	Y	Y	A	A	
Interactive aspects	Y	P	A		Y	Y	Y	Y		Y	Y	Y	Y	Y	Y	P	
Data collection via form		Y	A		Y	Y				P	P	Y	Y	Y	Y		
Data collection via form on client side		Y	A		Y	Y		Y				Y	Y	Y	Y		
Dynamic presentation/interaction generat.							Y	Y						Y	Y		
Multimodal interaction							P	P		P		P	P	A	A	P	
Production and Player Tools																	
Authoring Tools	P		Y	Y	Y	Y	Y	Y			P	P	Y	Y	Y		
Players: PC, Mobiles, PDA, STB, iPhone	Y	P	Y	P	Y	Y	Y	Y	P	P	Y	P	Y	Y	Y	P	P
Download distribution	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Streaming/Progressive download	Y	P			Y	Y		P	Y	Y		Y	P	Y	Y		
DRM/CAS IPR management	Y	Y				Y								Y	Y		
Protection models and servers	Y	P				Y							Y	Y	Y		

Table 1: Comparative analysis among multimedia and cross media solution, where: Y = supported, P = partially supported, A = supported by plug-in/external extensions

In this context, a particular relevance has the modeling of links among different elements to create direct access paths to the detailed elements of intelligent content, so as to allow also referring to them from outside, for example, from other objects. Direct access is also strongly relevant to the direct play of digital essences wrapped to binary coded and encrypted file formats, since the access has to be aligned to the chunks defined in agreement with the protection and streaming models.

As to the file-format, most formats are XML based (e.g., MPEG, SCORM, HTML) while most proprietary formats are mainly binary (e.g., Flash, macromedia). The openness of the format specification has to be guaranteed to be interoperable. Most of the standard formats are open and well documented, while others are controlled by patents. Some formats may support both XML and Binary such as AXMEDIS, MPEG-21, MPEG-4. The advantages of binary formats are mainly in the performance and file size. Packaged formats integrating several essences and data are a way to reduce the complexity perceived as such by the user (e.g., MXF, AXMEDIS). When complex packages are distributed, the binary model may limit the capabilities of distribution for the complexity of the model.

The possibility of navigating into the content structure, including links for detailed access to single elements/segments with a sort of reflection capabilities, is a fundamental step to present some degree of intelligence *and thus adapting the content behavior according to content structure and logic. This kind of reflection on structural aspects has to be accessible on scripting and decision parts of the object intelligence.*

Descriptors are typically included into the object model as knowledge and representation to allow indexing and searching content. The content models have to support the extension/addition of descriptors such as in MPEG-21, AXMEDIS. Descriptors included into content objects such as in KCO [30], ACEMEDIA [22], AXMEDIS [27], may be used for creating powerful indexing and classification models, thus allowing sophisticated query search of content on the basis of semantics. Descriptors are used to formally make accessible reasoning algorithm knowledge and information about:

- object structure in terms of RDF relationships, DOM model, such as: is part of, is aggregated by, is protected. These descriptors are processed by the above-mentioned reflection capabilities to navigate into the object structure and elements.
- identification codes such as ISBN, ISRC, etc., and persistent identifiers [21].
- metadata for general archival multilingual classification such as DC, OAI-ORE, MARC, TvAnyTime, etc.etc., are frequently used,
- production information, which are used to model the content status and the workflow aspects of the content production such as: versioning, creation date, activities to be done, processing performed logs, etc.
- conceptual information related to the internal aspects of content such as rhythm, genre, presence of objects in the scene, scene description, color dominance, etc. They are typically modeled via XML, RDF, MPEG-7, and may refer to taxonomical model and ontologies.

- IPR aspects, such as the potential rights and/or the associated rights with each given resource in the content and those for the whole cross media content. Rights can be formalized in a sort of licensing languages where the principal/beneficiary is not specified or may be described with some generic user profile.
- annotations which can be cross media, links and as well as behavior coding; thus creating new non-linear paths and/or new behaviors into the intelligent content. For example, a video can be annotated in a given point (time and space of a scene, with some ***direct links***) by an audio, a text, a video, etc.; annotation of images, audio, video, documents, 3D, animations, etc.; activation of behavior from annotations, etc. Therefore, also in this case, the possibility of accessing to the single element/segment of the complex content package is mandatory to attach/refer annotations to them. Therefore, additional knowledge/descriptors and non-linear paths can be created by and/or shared among users. The annotations in the AXMEDIS model are based on the Annotation of the MPEG-21 DIDL. An AXMEDIS annotation can be a complex content modeled as a cross media AXMEDIS object or a simple part of it, and at each annotation: assertions, anchors and descriptors can be associated.

Most of these descriptors are only useful on server side. The typical server side exploitation of descriptors is the indexing and search, may be semantic queries. Some of them can be profitably exploited by the behavioral capabilities of the intelligent content, for example, to know how the content is structure, which are its potential rights, which kind of annotations or relationships with other content are possible, which kind of semantic modeling is foreseen, which behavioral functionalities are accessible, etc.
The content intelligence may take into account descriptors and metadata to infer/determine content behavior dynamically. This kind of descriptor reflection may be used to search and take decisions according to the knowledge related to digital elements/essences and to the content structure. The annotations have to be considered descriptors and may be included into the object package by changing its structure or may be externally located. In both cases, the capability of processing descriptors and the direct access via links are an enabling technology.

Behavioral aspects have to deal with synchronizations, animations, functional aspects as the procedural coding activated by events, procedures associated with multimodal user actions, and therefore the so-called formalization of the ***Content Logic***. This aspect is frequently enabled into complex content models by adding the possibility of interpreting coded behavior via languages (java, JavaScript, action script, for HTML, AXMEDIS, and Flash) (Rule Interpreter in Figure 1). The accessed functionalities via coding could include communications capabilities, event managements, collaborative and coordinated activities, reflection capabilities, and processing capabilities, etc. **Processing capabilities** are needed to transform content and/or knowledge/descriptors; for example to: access, transcode, change, adapt, convert, save, decrypt. To this end, additional external; information may be needed, such as profiles and descriptors regarding user, device, context, network, etc. (not stored into the content structure but coming from the player/device). On such grounds, the formalization of ***Content Logic*** may take into account profiles and descriptors to adapt the behavior. In MXF format, these aspects are missing, since it has been defined as an interchange format and not as a file format to reach the final user players. Some semantic descriptive capabilities of devices may be formally defined in CCPP [34] standard and also in the MPEG-21 DIA (digital item adaptation) [35] standard which formalizes some profiles and descriptors. In flash, the Content Logic aspects are formalized in Action Script. In MPEG-21, the content logic formalization is delegated to DIP/DIM [36] parts, which are not fully defined in terms of semantics in the standard. AXMEDIS largely extended the capabilities of MPEG-21 DIP and DIA with the adoption of JavaScript [37] and reflection into object structure, metadata and descriptors, and with several media and data processing capabilities, communication capabilities, profiling ingestion and processing, taking decisions, etc. Communication capability of intelligent content may support the simple communication with server or more complex peer communication among other intelligent content items. MPEG-21 has no support on these aspects, while AXMEDIS and Java may exploit this capability into the content logic as well. Moreover, the capability related to decision taking on the basis of current profiles and content status implies a certain level of autonomy directly enforced into the content object via a Behavior Engine. This kind of autonomy can be enforced in terms of rules (coded via procedural languages) and with some semantic inferences on the basis of ontologies and RDF modeling. Examples of the usage of content logic may be the capabilities of making internal: content search, content self-adaptation, dynamic production of GUI interface according to the user capabilities, exploitation of new user devices, augmentation of collected knowledge and information into the object structure, etc.

Therefore, the content intelligence is enabled by behavioral capabilities by which the single content may present some autonomous behavior in reacting to contextual stimulus, user profile, user actions, content structure and descriptor analysis, and reasoning, etc.

Presentational and interaction aspects are related to the user interface. These activities can be performed on the presentation model, for example on HTML, SMIL. Interaction aspects are typically integrated into presentation models via buttons, active regions, input forms, etc. HTML provides forms to collect data and send them to server side. SMIL is not focused on the data collection from the user, while it has some active regions, layouting, animations. These formalisms have already some direct capabilities to adapt the code to the context (such as language, browser, etc.), while more elaborated solutions are needed when these models are used into cross media content packages such as MPEG-21, AXMEDIS, and MHP. The presentational and interactive possibilities may be totally pre-encoded such as in MPEG-4, MHP, etc., or may be dynamically changed generating dynamically the presentation layer.

The intelligent content may need to adapt user interface and therefore already available content features according to user profiles, device descriptors, etc. The technical activities can be: repurposing, reformatting, and adaptation. They have the capabilities of activating

JavaScript/java functional aspects, so that the presentation may be controlled by some behavioral coding. By means of the presentation/interaction layer, the *Content Logic* may request to users to collect new information to change behavior according to internal context and descriptors; for example, preferred font size, preferred language, dominated colors, image size, etc. This means that, the collection of data via forms has to be closed on the client side and not via server as in HTML. To this end, simple MPEG-21 DIP solutions are not viable and a full form redirection of Content Logic call back has to be provided as in Flash, MHP, AXMEDIS with HTML, AXMEDIS with SMIL.

Recently, multimodal user interactions are requested to support different devices (mouse, remote control, mic, gesture, joystick, etc.), external events and actuators, interaction with 3D represented environments, etc. The presence of processing and behavior aspects may help in creating more powerful interactivity on client side.

Therefore, presentational and interactive capabilities are useful to support the intelligent behavior of content when they are integrated as direct functionalities of content logic. Main functionalities are related to dynamic production of user interface, getting data from the user, activating call back from the user interface events, adding functionalities to the content logic, for example those coming from haptic devices, GPS location, barcode readers, and external devices and context.

Authoring and players for intelligent content may be classified into:

- simple manual authoring tools which can cover only some aspects (see for example the MPEG-4 authoring tools).
- complete authoring tools endowed of graphic user interface and help to support powerful products such as Flash, HTML, SMIL. Advanced authoring tools may provide support for versioning and undo, if the content model supports them.
- automated production tools supporting the massive creation and repurposing of content such as some specific format tools as the Adobe flash tools, or as general purpose media grid tools as AXMEDIS AXCP which may cope with the production of a range of formats: NewsML, MXF, MPEG-21, AXMEDIS, HTML, XML, SMIL, MPEG-4, etc.

For the client side (see Figure 1), it is quite widespread to distribute simple players that allow playing a limited number of content formats (via a set of renderers and codecs) without any or with strongly reduced capabilities of authoring – e.g., flash player, SMIL, AXMEDIS player. Players may be provided for multiple platforms such as: PC, PDA, mobiles, iPhone and STB/decoders.

The authoring tools and players may have support for the protection and enforcement of the IPR (intellectually property rights). Simple solutions are based on CAS, conditional access systems, which control the access to the content. Typical broadcasting distributions of MHP content (e.g., DVB-T in the present implementations in Europe) and of MPEG-4 such as in OpenSKY are based on CAS; they have limitations in the number of business models they may support and implement. More sophisticated technologies are based on the so-called DRM (Digital Rights Management) which typically presents integrated protection technologies to keep under control the consumption/enforcement of rights into the players and may be into the authoring tools, according to formal semantics of rights. The DRM may support different business models such as pay per play (right, extract, compose, distribute, etc.), subscription, etc., with conditions on time, location, execution times, ... Some of the models are formalized via licenses in terms of REL (rights expression languages) such as that of MPEG-21 REL [38] and OMA Open Mobile Alliance DRM [39]. Therefore, content players and authoring tools have to enforce the DRM/CAS to respect IPR models. Some players have also to provide support for the real-time distribution protocols such as streaming, progressive download. When complex cross media content is taken into account, the protection can be inhomogeneous on each and every resource, and real time decryption and play lead to a complexity very hard to be managed, and not free from risks. Therefore, most of the content formats supporting DRM are mainly focused on simple download, while only a part of them can be streamed or progressively downloaded, while being played. Typically, only simple formats, that can be streamed for single audio visual items, support protection, since efficient players with real time play capabilities of protected content with DRM/CAS are quite complex to be realized.

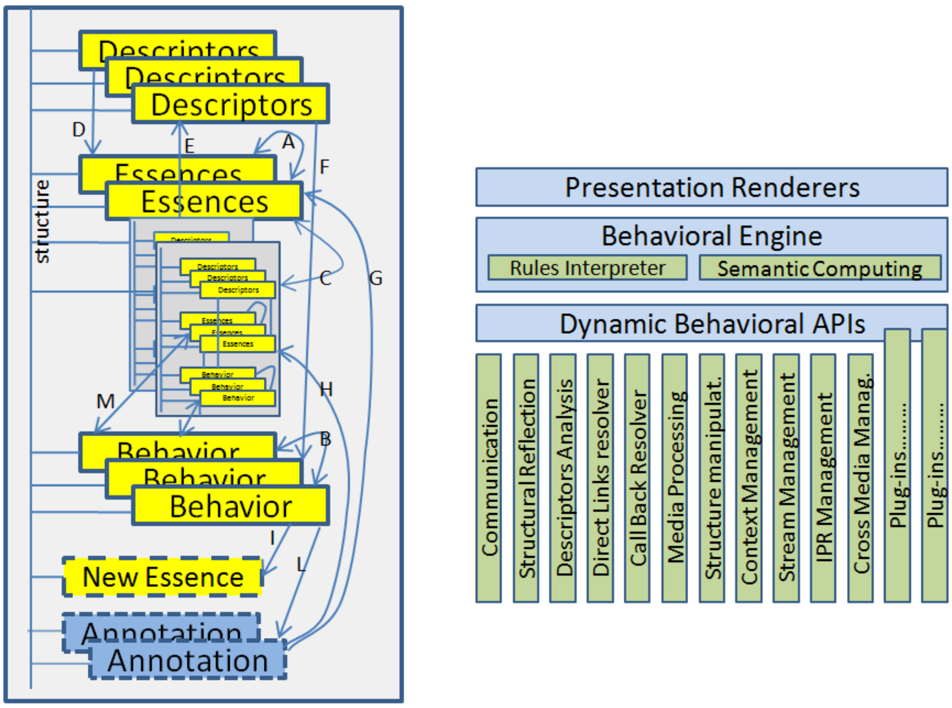


Figure 1 – General Intelligent Content Model and Player. Some examples of relationships: A: links among essences such as HTML, B: procedure calls, C: essences linking to other nested essences, D: content descriptors, E: complex objects accessing to descriptor information. F: descriptor of behavioral aspects, G: annotation of an internal essence or segment of it, H: annotation to nested essences, I: production and activation of essences, L: activation of an annotation, M: adaptation of a nested essence.

The player reports only the functionalities while the underling model is accessed via the several API for accessing, navigating, understanding, manipulating and producing the model elements of the intelligent content.

The performed analysis brought us to a very complex and articulated scenario, since each column of the table implied the assessment and verification of several aspects. Therefore, a wide range of models and formats has been reviewed and for each model and format we gave a full Yes only when the corresponding feature was almost satisfactory and able to cover most of the issues mentioned in the table description. On the contrary, the table cell has been left empty. In some cases, the considered features have been identified as Partially covered by the provided solution, so that a P has been assigned. In some other cases, when the features are covered by additional tools, an A has been placed.

3. Main Aspects of AXMEDIS Intelligent Content

The AXMEDIS format has a set of capabilities , which allows its usage as intelligent content. AXMEDIS format extended the MPEG-21 format with a set of features, which can be mainly listed as: file format, package, extended data and semantic processing, metadata, descriptors, and tools. The main technologies to enforce intelligence into AXMEDIS content are the:

- descriptors (metadata, annotation, structural description, profiles, etc.)
- behavioral formalization (processing capabilities, reflection capabilities).

Both features are included into structural aspect modeled format by using the MPEG-21 DI, DIDL. According to AXMEDIS any kind of digital essences/files can be enforced into the package including: metadata, identification codes, presentation information, descriptors, annotations, and JavaScript methods called AxMethods. Among the general descriptors, the Dublin Core and the so-called AXInfo are included by default. The AXInfo is a set of metadata used to manage the AXMEDIS life-cycle (Object Creator, Contributors, Object Owner, Object Status, Potential Available Rights, PAR, that can be acquired on the content, etc.). The AXMEDIS model is extensible, since it allows adding any kind of additional files/essences and descriptors.

AXMEDIS model and format support the DRM and protection with authentication and certification of players. AXMEDIS protection model fully supports the AXMEDIS complex structure with nesting levels and it can support multiple and nested protection models. AXMEDIS DRM allows defining licenses with multiple rights and conditions according to MPEG-21 REL standard. Figure 2 provides a sketch of the content transformation after a protection process in terms of MPEG-21 elements. To make the figure simpler and more immediate, only the relevant MPEG-21 elements have been considered/ depicted. The Descriptor element, as an example, is not a leaf since it can contain a Statement as its child element. The binary protected version of AXMEDIS objects is optimized and coded according to the ISOMEDIA standard. The File format of AXMEDIS can be also XML.

The diagram highlights which parts of the content structure remains accessible even after the protection. Please note that some *Descriptor* elements have been tagged with the “public” attribute. The “public” attribute has been introduced by AXMEDIS and it allows specifying the `Pub-desc` of the model. The concept can be explained as follows: even if the root level of the AXMEDIS Object hierarchy is protected, all the “public” metadata are copied on the appropriate *ContentInfo* element, which has been designed by MPEG-21 IPMP to contain description about the protected content according to AXMEDIS input requirements. Since the content structure is hierarchical, it has been decided to mimic the content structure and use MPEG-21 DIDL hierarchical element Item. AXMEDIS authoring tool can apply multiple protection algorithms on each content element [10], [28].

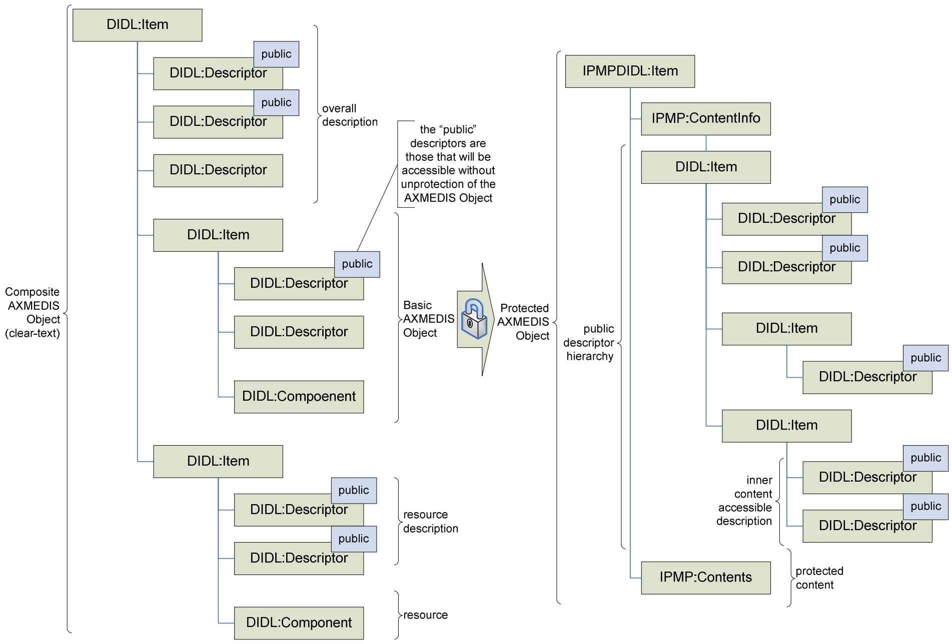


Figure 2 – AXMEDIS Object after protection in terms of MPEG-21 elements

AXMEDIS distribution capabilities have been particularly developed. AXMEDIS content can be downloaded, streamed or progressively downloaded. For the download, AXMEDIS has no limitations, while for the real time streaming and/or progressive download the AXMEDIS objects have to be mainly audiovisual. For example, on PC or mobile the AXMEDIS objects containing metadata, descriptors, presentation layer and an audiovisual essence can be streamed and progressively downloaded. Therefore, AXMEDIS objects with simple SMIL and audiovisual can be also played in real time on the AXMEDIS Mobile and PC player. On mobile devices, the AXMEDIS complex content with nesting levels can be decomposed into multiple objects to be played while distributing. In all these cases, the AXMEDIS DRM based on MPEG-21 REL is working [38], [28].

In AXMEDIS, the behavioral aspects (i.e., Content Logic) are activated and formalized in terms of AxMethods, which are coded in extended JavaScript; they extended the MPEG-21 DIP/DIM standard [36]. In MPEG-21, the semantics of DIP/DIM is not enough to cover all the aspects described in this paper and it is limited to basic functionalities [40]. The reflection capabilities on the object structure enforced into AxMethods allow the navigation, the creation of nesting levels and the direct access to the resources via direct links and references. The AxMethods may navigate in the object structure and into any kind of object descriptors including the ones formalized in XML, RDF, TXT included into the object structure. This allows the coding to exploit the knowledge and representation to enforce intelligence, and even to activate decision taking and inference engines. Descriptors may be freely included into the AXMEDIS content package at any nesting level and referred to internal or external essences. On such grounds, the indexing and the semantics search can be performed also on these content formats and models. In AXMEDIS, processing capabilities are an integrated part of the behavior and they are inherited from the back office AXMEDIS Content Processing, AXCP, language [21]. It enforces capabilities of semantic computing, and model reflection, communication, event managements, data and descriptors processing, profile processing, taking decision engine, rendering, user interaction, indexing and queries, etc.

3.1. *Dynamic behavior for presentation layers*

AXMEDIS content processing capabilities support the management of multiple presentational models in the same package. The same methods can dynamically create and/or modify existent packaged content to adapt them, to produce a new user interface, an adapted form, etc. This allows any dynamically adapting of the object behavior on the basis of the user profile, device profile, GPS position, user inputs, past actions, content descriptors, etc.

AXMEDIS content may provide an intelligent behavior on the basis of the current profiles (user, device, context, network) and content status with autonomy in the decision making process directly enforced into the content object. For example, it may alternate HTML, SMIL, video/images and FLASH animations in dominating the main canvas of the AXMEDIS player. This aspect is not dependent on MPEG-21. HTML and other formats may have dependencies with respect to style sheets, such as CSS, and digital essences (text, video, audio, image, etc.) directly hosted into the AXMEDIS package. In AXMEDIS tools, presentation layers such as SMIL, HTML, and FLASH may put in execution AXMethods for activating behavioral actions allowing to inspect and modify the content structure (e.g., add new resources), control the resource rendering, perform calls to web services, etc. Therefore, the interactivity of AXMEDIS objects can be closed on server side with direct call and/or directly into the object by activating some AxMethod call back. This means that forms and/or buttons may activate AxMethod to process data and play other content from AXMEDIS package itself. This allows to pass parameters from one essence to another and to create sophisticated intelligent content behaviors. To this end, it is possible to use relative URLs (for resources and methods contained into the same object) or absolute URLs. An example of relative URL for an AxMethod is: axmethod/methodname(). Absolute URLs can be composed as:

axmedis://AXOID/resource-localpath (resource url)
axmedis://AXOID/axmethod/methodname() (axmethod url)

In the above example, it is assumed that the resources/methods are contained into the root object. In the presence of nested AXMEDIS objects, it is possible to refer to resources/methods included in the child objects containing the method by using:

axmedis://AXOID//axoid-of-the-child/resource-localpath (resource url)
axmedis://AXOID//axoid-of-the-child/axmethod/methodname() (axmethod url)

Please note that, the AXOID is the unique ID that allows the identification of the single object. If more than one object has the same AXOID, the first of the AXMEDIS package is taken. The above-described capabilities may generate new versions of the digital essences into the AXMEDIS object or may create other AXMEDIS objects. On the basis of the above features it is evident that the proposed AXMEDIS model allows the creation of Evolving Intelligent Content. The capability of evolving is not present in other models mentioned in the introduction. In the presentation layer, it is possible to link a specific element of HTML presentation by another HTML presentation.

3.2. *Examples of Intelligent Content*

Several examples of intelligent content can be accessed on the AXMEDIS portal. The recently issued version of the AXMEDIS player includes the possibility of executing AXMethods as described above. Some examples are provided hereafter.

The first example provides a cross media intelligent content with several tools for estimating medical dosages of more than 30 different medicines (see Figure 3), see Mobile Medicine http://mobmed.axmedis.org for other examples. Each medical dosage has a complex description including dosages, category, counter indication, usage, indications, prescriptions, etc. Each medical tool can be accessed by means of the menu provided in the first page, while the user may perform an internal search into the content to look for any specific tool/medicine addressing a certain problem.

The internal content search capability has been implemented as AxMethod as depicted in the following JavaScript code. It can be executed directly on the mobile without the needs of having a complex player since the content contains both descriptors and behavior.

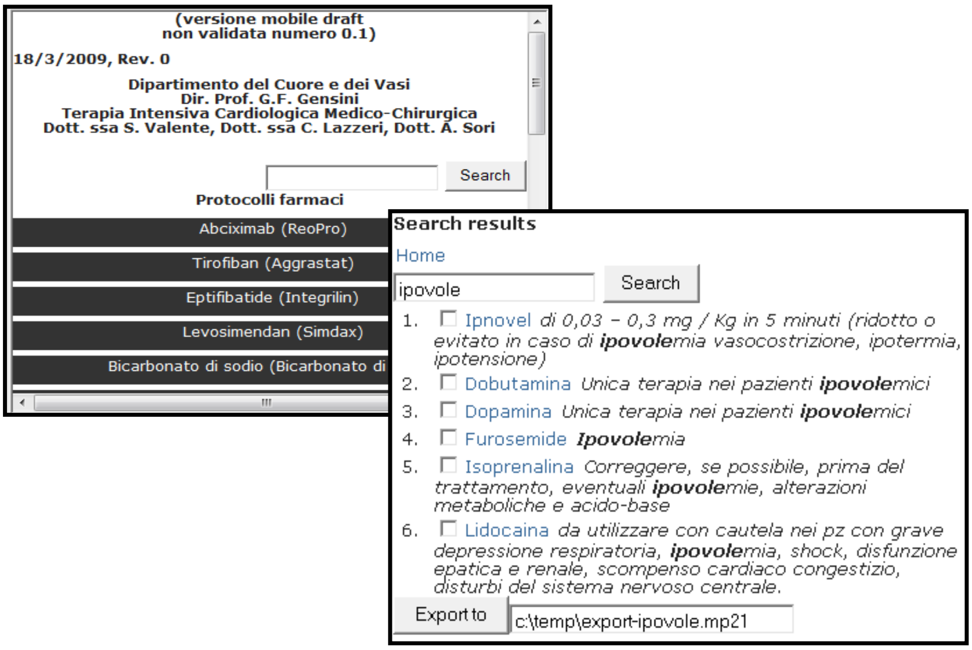


Figure 3 – An intelligent content, demonstrating capabilities of reflection, dynamic interface production, self search, self modification/adaptation, content migration, etc.

A large range of possible searching algorithms can be realized enforcing (even on mobile) a certain level of intelligence. Complex reasoning can be implemented by changing the search functionality, for example by activating a taxonomical/ontological search. This also depends on the player capabilities. In the case of http://mobmed.axmedis.org more complex semantic queries can be performed into the content organizer and directly on the portal which provides on the basis of content descriptors some semantic computing functionalities: search, suggestions.

```
function search() {
 var s=URL_QUERY_ARRAY['search'].replace(/\+/g,' ');
 s=s.replace(/^\s*|\s(?=\s)|\s*$/g, "");
 var html="<html><head><link href='style.css' rel='stylesheet'
         type='text/css' media='screen' /></head><body>";
 html+="<p><b>Search results</b></p>"
 html+="<a href='index.html'>Home</a>";
 html+="<form action='axmethod/search()' method='get'>";
 html+="  <input type='text' value='"+s+"' name='search'>";
 html+="  <input type='submit' value='Search'>";
 html+="</form><ol>";
 if(s.length>0)  {
  var c=axDocument.getContent();
  for(x in c) {
```

```
    if (c[x] instanceof AxResource && c[x].localPath!="index.html" &&
      c[x].localPath!="search-results.html") {
     var t=c[x].toString();
     var p=t.toLowerCase().indexOf(s);
     if(p>=0) {
      var title=extractMetadataDescription(t);
      html += "<li><a href='"+c[x].localPath+"'>"+title+"</a>
              <em>"+extractDescriptors(t,p,s)+"</em></li>";
     }   }   }   }
 html+="</ol></body></html>";
 var rs=new AxResource;
 rs.loadFromString(html,"search-results.html","text/html");
 rs=axDocument.addContent(rs);
 DIP.play(rs,true);
}
```

The AxMethod “axmethod/search()” is activated by the button presented in the main frontal HTML page of the object (see Figure 3), and into the result page that is automatically generated, see form action='axmethod/search()'. The AxMethod navigates in the axDocument structure accessing to package resources and it generates dynamically a new web page called “search-results.html” with search results; such page is stored into the object package as a new AxResource, exploiting the MPEG-21 DIP standard call. The produced web page also presents links to any single resource of the package via the local paths. It provides the user with the possibility of selecting and extracting some medical information to produce a derived content collection. The following method “build()” uses the reflection capabilities on content structure, metadata, descriptors to generate derived content items, adapting content, presentation, behavioral aspects.

```
function build() {
…..omissis…..
 var axobj=new AxmedisObject();
 var r=new AxResource;
 r.loadFromString(html,"index.html","text/html");
 axobj.insertContent(r,axobj.getContent()[0],true);
 axobj.addContent(axDocument.getResource('style.css'));
…..omissis…..
 var dcFrom=axDocument.getDublinCore();
 var dcTo=axobj.getDublinCore();
 dcTo.addDCElement("title", "Derived from "+
                   dcFrom.getDCElementValue("title"));
 dcTo.addDCElement("creator",dcFrom.getDCElementValue("creator"));
 dcTo.addDCElement("subject",dcFrom.getDCElementValue("subject"));
```

```
…..omissis…..
  axobj.save(file);
  DIP.alert("Exported object can be found in "+file,1);
  DIP.play(axDocument.getResource('index.html'),true);
}
```

The second example is an AXMEDIS object containing a collection of other AXMEDIS objects, which may be news, see Figure 4.; in this specific case, each of these nested AXMEDIS objects has pieces of information regarding an actor such as: DC metadata, biography, short video, descriptors, html page, etc. The AXMethods included into the collection object allows navigating into the collection to search in the HTML and descriptors of the collected objects in order to make a selection of them.

The search is coded by using the AXMethod, which is activated by an HTML page; it performs the search in the nesting levels and creates an HTML page to produce the results as depicted in the following code. The results produced by the search have been included into the original AXMEDIS object as a new HTML page and a digital resource of the package (see Figure 4 on the right side) thus creating an evolved intelligent content, or they can be used to generate a new AXMEDIS object. The example provided can be improved to take into account user's preferences, context, etc., in providing the results and/or in saving the produced results.

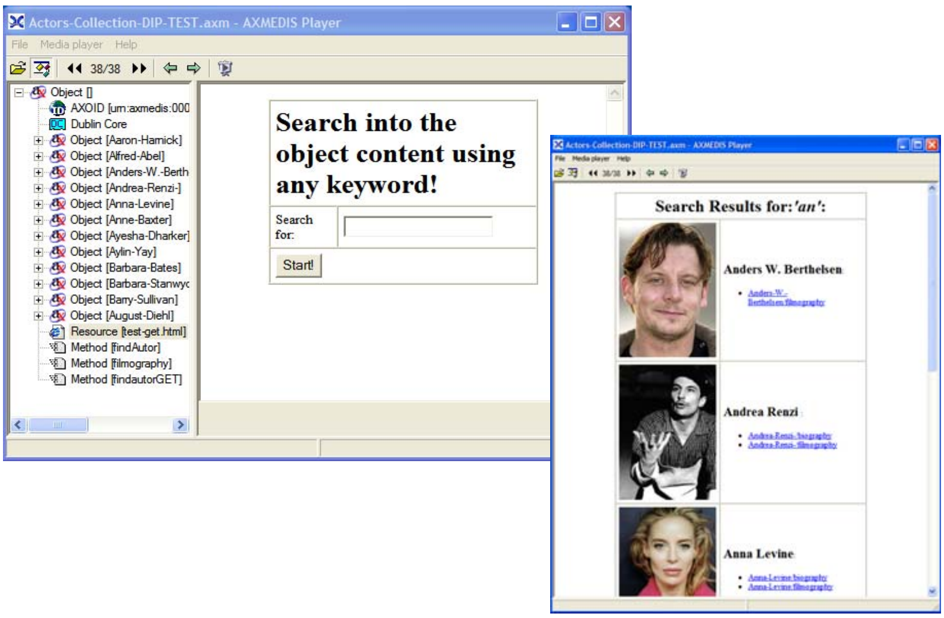


Figure 4 – Collection of nested content

Another intelligent content may help the user to protect and produce licenses for its users. In this case, the intelligent content implements the concept of license maker by requesting the needed information to the user, protecting the content, issuing licenses to enable a list of users to play the content for a limited number of times in a temporal window (see Figure 5). In this case, the intelligent content is managing a database of produced objects and a list of users, and it issues the license by posting information on the AXMEDIS DRM server. This version of license maker for content distributor may be generated by a producer license maker. Both of them are accessible on the AXMEDIS portals for any possible perusal. With this tool, a journalist may produce news, protect news, and distribute them via P2P or post on public portals and provide access to them only to authorized people and institutions.

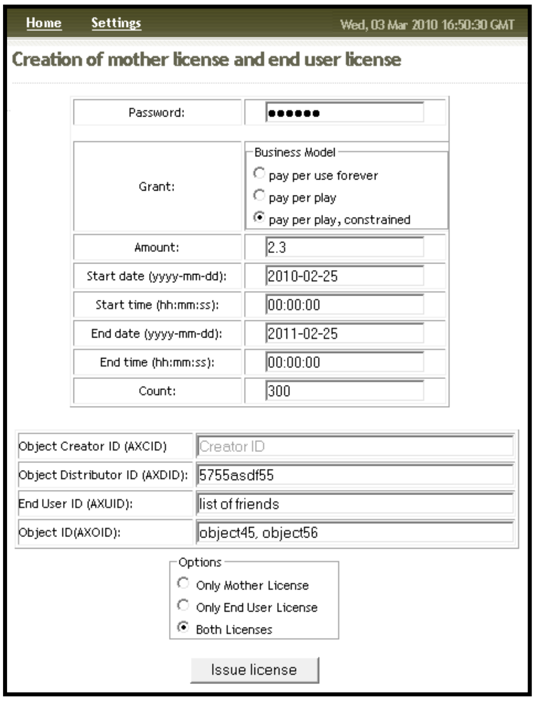


Figure 5 – Collection of nested content

Another intelligent content may help the user to protect and produce licenses for its users. In this case, the intelligent content implements the concept of *license maker* by requesting the needed information to the user, protecting the content, issuing licenses to enable a list of users to play the content for a limited number of times in a temporal window (see Figure 5). In this case, the intelligent content manage an internal database of produced objects and a list of users. On this basis, it may issue the licenses by posting specific information on the AXMEDIS DRM server (see Figure 9). When the content is produced and licensed to be distributed by third party people, this version of license maker for content distributor may be generated by a *producer license maker*, which can issue licenses to authorize the distribution of content. With this tool, a journalist producing news, may protect is own news, and distribute them via P2P or post on public portals and provide access to them only to authorized people and institutions. Both of these intelligent content for licensing are accessible on the AXMEDIS portals for any possible perusal. Professional versions may support the internal search of licenses descriptors.

Another example of intelligent content can be the solution adopted by Digichannel audio-visual content portal (http://digichannel.net/). It distributes audio, video, photo, books as packaged content into MPEG-21 intelligent objects, also protected via AXMEDIS DRM. The content can contain multiple and different items such as a CD ISO image plus cover and other pieces of info; or the DVD ISO image plus a booklet and images. The content itself is a wizard helping the user to understand what is inside and what he may try, and finally providing support and offering feedback to acquire a license to see the content automatically unpacked on the user device. The last example can be the solution proposed in AX4HOME [46] of BBC. In that case, a tool or a content may help the user to perform the recording of free on air video transmission in DVB-T. Once the recording is performed, a protected content package is produced collecting other information from the P2P network. The resulted object, may present the recorded video, additional content such as: video, biographies, images, games, and links to commercials and promotions. Moreover, the content package can be automatically protected to be make it usable only on the user house, that is its DRM domain.

4. From NewsML to AXMEDIS modeling passing via MPEG-21

Among the formats mentioned in the introduction, the ones used for distributing and sharing news are mainly text and XML oriented such as NewsML of IPTC (International Press Tele-communication Council). Recently a new version of NewML has been proposed, namely the NewsML-G2 providing support for both referencing textual news, resource files, and paging them, while collecting metadata and descriptors, vocabularies, etc. (http://www.iptc.org). Furthermore, the news are typically massively processed by news agencies and/or TV news redactions. They are received in NewsML formats, and as well as in many other formats such as: HTML, plain TXT, PDF, and RTF. The agencies and redactions need to move, transcode, and adapt them to different formats, thus processing both text and digital essences, by changing resolution, summarizing text, adapting descriptive metadata, etc. In some cases, the adaptation has to be performed on demand as a result of an answer to a query or request to a database or on a web service.

Moreover, frequently the news may contain videos and images, while the solution proposed by NewsML, which is to zip the file, forces the user to unzip the files in some directory in order to access and play the video. Besides, news may contain sensitive data which protection of content or IPR (intellectual property rights) is asked for. Therefore, most of the cross media formats mentioned in the introduction present a certain number of problems in modeling news, which means limitations related to the adopted packaging format. For example, the very relevant limitation of NewsML format and package does not allow the user to:

- perform a direct play of the video content without unpacking/decompressing, with related limitations,
- protect the news content over distribution,

- preserve the IPR (intellectual property rights),
- enrich the news with additional information, while keeping separate the augmented information and the core parts.
- distribute a collection of news by making internal search, without decompressing and un-packaging it.
- associate each news with specific intelligent functionalities, which are only accessible for the manipulation of that specific news.

Such problems are strictly related to the file format and subsequently to the protection support including certification, content signature and licensing. For this reason, the usage of AXMEDIS/MPEG-21 format has been considered as a solution to cope with these problems.

In further details, the NewsML format has a structure at 4 nested levels (from the contained to the smaller components): *NewsEnvelope*, *NewsItem*, *NewsComponent* e *ContentItem* (*http://www.iptc.org*). The *News Component* mainly contains the information that may be used for modeling the *NewsItems*. At the end, the *ContentItem* describes the contribution in terms of comments, classification, media type, format, notation, etc. The NewsML has also metadata mapped in the architecture and in particular in the *NewsComponent*: Administrative Metadata, Descriptive Metadata, and Rights Metadata (as a simple declaration that does not enforce the content protection). The information for the news identification is reported into the *NewsItems*, each of them can be univocally identified. On the basis of our analysis, we have identified 6 main entities which have to be addressed: NewsML, NewsItem, NewsComponent, ContentItem, TopicSet, Catalog (see Figure 6).

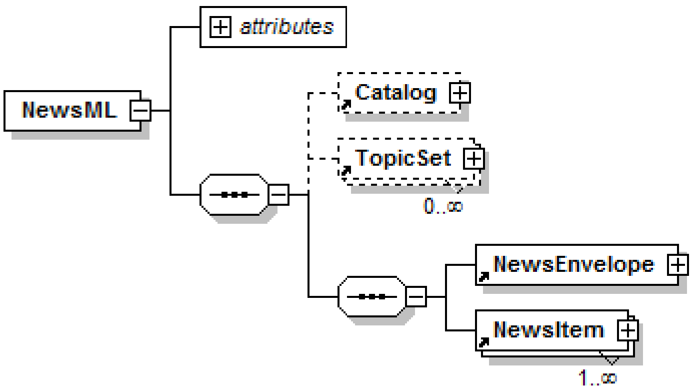


Figure 6 – NewsML main entities

The general model of NewsML is hierarchical, and thus this complexity have to be taken into account, in the ingestion, analysis and conversion. To this purpose, the NewsML model has been replicated into an object-oriented modeling allowing us to represent it in the memory (before analysis and conversion), by considering their relationships and roles, as in the UML diagram reported in Figure 7. Furthermore**,** other classes have been implemented to model the NewsML, namely: Topic, NewsMLDocument, NewsComponent, NewsItem also specialized from both NewsMLElements and ContentAttribute. The proposed model allows to ingest quickly the NewsML structures.

The realized model allows the efficient performance of such needed transformations on the NewsML files. For example, the extraction of a NewsComponents removing its parts from the tree, the addition of news, etc, together with the conversions of the NewsML in other formats such as XML, HTML, Text and files, and MPEG-21 as described hereafter.

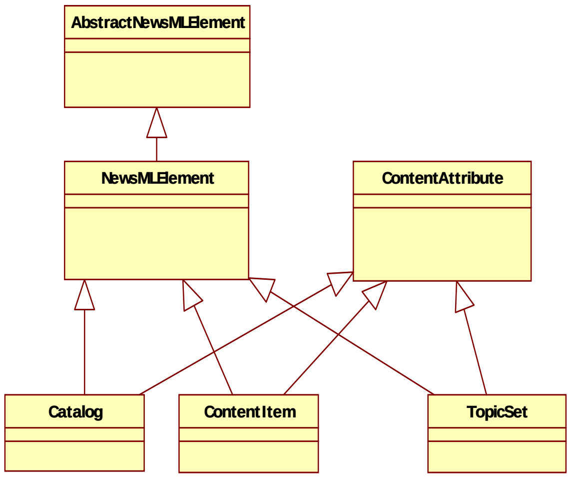


Figure 7 – Modeling NewsML main entities for conversion and analysis

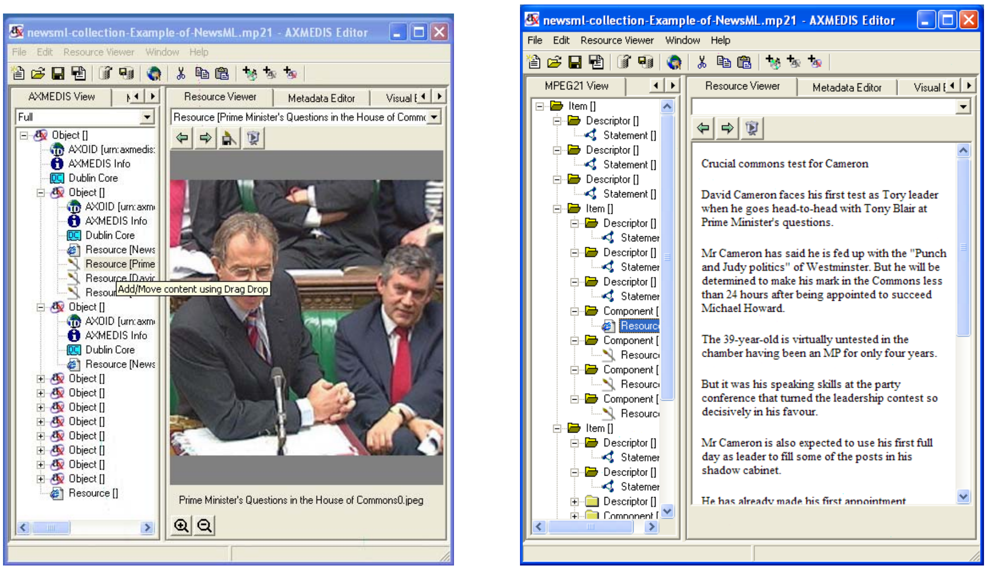
The resulted model has been also analyzed to map the information into the MPEG-21 structure of the DIDL (Digital Item Description Language).

Table 2 provides a Mapping of the NewsML elements with those of MPEG-21 and AXMEDIS. Please note that AxMethods does not present a counterpart into the NewML, while the DIP/DIM standard is not comparable with their expressivity.

NewsML element	AXMEDIS Element	MPEG-21
Metadata	AXInfo + Dublin Core	Descriptor
Metadata	Descriptors	Descriptor
NewsML	AxObject	Item
NewsItem	AxObject	Item
NewsComponent	AxResource	Component
ContentItem	AxResource	Component
--	AxMethods	(DIP/DIM)

Table 2 – Mapping concepts of NewsML to AXMEDIS and MPEG-21

The AXMEDIS editor allows the user to see both MPEG-21 and AXMEDIS views of the NewsML file as depicted in Figure 8. AXMEDIS view is simply a more abstract view of the AXMEDIS file format based on ISOMEDIA. The AXMEDIS mapping is more effective and easy to understand than the underlining MPEG-21 modeling, which is fully flat and difficult to be understood by humans. The resulting MPEG-21 container of the News can be protected by using AXMEDIS DRM/MPEG-21 REL tools. In Figure 8, an AXMEDIS view where the nesting levels of AXMEDIS objects are self-evident. They can be moved or extracted simply by dragging and dropping. The same approach can be adopted to work with single contributions: text and/or digital files (images, video, etc.). They can be played directly into the editor and into the AXMEDIS player. An additional feature is the index in HTML of the converted NewsML items. It has been automatically produced by processing the NewsML structure in the AXCP script. That index is an HTML file enforced into the AXMEDIS Object (see the tree in Figure 8, on left).



AXMEDIS view and editor **MPEG-21 view and editor**

Figure 8 – A NewsML on the AXMEDIS Editor, editing AXMEDIS and MPEG-21 models.

4.1. *Automated Ingestion and Repurposing of News*

The above-mentioned object oriented module for NewsML ingestion, modelling and processing has been added to the AXMEDIS framework. Therefore, a set of functionalities, APIs, to access and manipulate the NewsML models has been made accessible directly into the extended AXMEDIS Java Script language for the multimedia processing used into AXMEDIS Content Processing Media grid [41], [42], (AXCP), and also into the AXMEDIS Editor. The AXCP processing capabilities accessible from the AXCP rules allow to perform activities of ingestion, query and retrieval, storage, adaptation, extraction of descriptors, transcoding, synchronisation, fingerprint, indexing, summarization, metadata manipulation and mapping via XSLT, packaging, protection and licensing in MPEG-21 and OMA, publication and distribution via traditional and P2P channels, processing and navigation on descriptors and content provided in more than 150 different formats among them: XML/DOM, HTML, TXT, SMIL, MPEG-21, MPEG-4, MPEG-7, DC, OAI, CSS, XSLT, associative arrays, regular expressions, many videos, audios, images, documents formats, see AXCP technical note on [26]. The semantic grid and media grid are new solutions that are presently becoming largely diffuse in the back offices, see for example [44], [45]:

The AXCP architecture is substantially a media grid infrastructure made of a server cluster that includes two or more Rule Schedulers in failover configuration and several AXCP Node executors for executing Rules on separate computer systems (see Figure 9). AXCP Rules are formalized in AXCP JavaScript [42], and AXCP grid visual programming [43]. The Scheduler performs the rule firing, node discovering and any possible problem management and revering from failure. The Scheduler may receive commands (to invoke a specific rule with some parameters) and provide reporting information (e.g. notifications, exceptions, logs, etc…) to external workflow and tools by means of a WEB service. Requests may arrive from the general workflow management system implemented into the press agency and/or into the back office of a redaction. Compatible workflow management systems are BizTalk and OpenFlow.

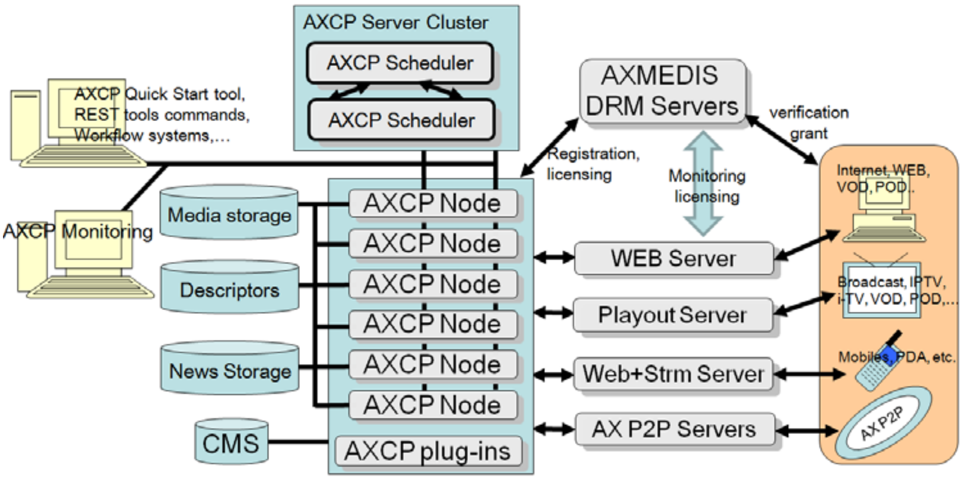


Figure 9 – Architecture for massive news ingestion, repurposing, and multichannel distribution. News are received via upload (a sort of UGC), email, ftp, sftp, P2P. They are automatically ingested via the AXCP Nodes.

Furthermore, each Rule Executor could invoke the execution of other Rules on the grid by sending a request to the Scheduler. This may be particularly useful in order to divide a complex Rule/procedure into sub rules/procedures running in parallel, thus allowing a rational use of the computational resources accessible in the content factory, on the grid. This solution preserves the advantages of a unified solution and it allows enhancing the capabilities and the scalability of the AXCP solution.

The AXCP processing tools are supported by a Plugin technology which allows each AXCP Rule Executor to link dynamically any content processing tool and algorithm (e.g. audio, video and image adaptation, transcoding, encryption) and to cope with possible customized algorithms and tools.

This solution based on AXCP allowed to set up flexible automatic processes where news information (coming as NewsML, HTML, TXT, ZIP formats) is ingested and processed in a very efficient manner; the solution has considered any kind of conditions and structures for repurposing news information and adapting them, text and digital essences included, into different formats: HTML, TXT, PDF, MPEG-21, SMIL, etc., either by integrating or not digital essences into them and distributing them via email, posting on FTP, on DBs, etc. In this context it is also possible to redistribute news towards a set of possible front ends such as: Web Servers, play out streaming servers, mobile distribution servers, and P2P servers.

4.2. *Advantages in using AXMEDIS to model cross media news*

Besides, any news modeling with AXMEDIS format and AXCP processing has the following advantages:

- t**he news modeled as AXMEDIS intelligent content** can:
  - be proactive, which means to request to their consumers/professionals to add their opinions, maybe a video recording, interviews, etc., which can be used to enrich and complete the cross media object. The completion can be planned and verified via reflection.
  - be active, thus allowing the user to search into a collection of news (e.g., several different versions or several different news) and extract them directly on the player by exploiting the intelligent capabilities of the content which regenerates the user interface and a new consistent content package with a selection of elements as presented in this paper.
  - be used as a news descriptor and/or a news container supporting any kind of file formats for the digital essences into the news.
  - be searched into the internal body of the news segments, thus making the understanding and browsing of complex news easier, by adding simple intelligent methods such as those presented in this paper.
  - be annotated according to AXMEDIS/MPEG-21 as described in [5]. Annotations can create non-linear story paths into the object package among its resources or among a collection of internal/external resources/objects.
  - be distributed in several ways and accessed via PC, PDA, iPhone, etc.
  - be protected when the information is distributed towards unprotected channels or when it contains sensitive information. The protection can be supported by DRM and by a plethora of different players for PC, PDA, and mobiles.

   - be collected, organized and searched taxonomically into the mobile devices (such as: PDA windows mobiles, iPhone, iPod, iPad). This is possible thanks to a recently distributed tool called AxObjectFinder which presents a certain degree of intelligence providing local recommendations and tracking user behavior on mobile phone [5]. AxObjectFinder is available for PDA and iPhone (the latter is called Mobile Medicine in AppStore).
   - provide adapting graphic user interface according to the user profile, preferences, device capabilities, etc.
   - include capabilities to protect and license content. And in particular, to protect/license the current or other user generated content objects.
- **the AXCP tools** can be used for automated:
   - ingestion of news which may arrive in several different formats (NewsML, XML, HTML, SportML, TXT,..), and may contain in the ZIP or refer media essences. In most cases, the ingestion includes the extraction of video resources from ZIP and/or from interchange formats such as MXF. Moreover, some news may even containing MXF video/audio that have to be decomposed in text, metadata, descriptors and video/audio files; then the audiovisual items have to be adapted for preview and distribution into the press agency;
   - repurposing towards different formats. For example for the news reproduction on titlers by using some Character Generator (for example using MOS).
   - redistribution on different channels and devices.
- **the AXMEDIS editor and player tools** can be used to:
   - preview the media elements into the news,
   - add other pieces of information to enrich and/or evolve the intelligent content, by continuously verifying its consistency, correcting and completing descriptors, requesting missing information to users. The evolution, may imply to change the content, for example, adding resources, adding comments, correcting information, adding descriptors, merging/updating/completing news, creating connection and references to other news, etc.
   - make a direct play of the essences into the news, without extracting them from the package. In fact, news are frequently distributed as ZIP files that have to be decompressed to get access to each single resource.

Most of the features described above are not only interesting for news. Similar problems of widespread generation, annotation, self protection, etc., are also suitable for the User Generated Content, UGC.

5. Conclusions

In this paper, the Extended AXMEDIS model and format as intelligent content and format has been presented. The new format and tools have been improved/extended with a large set of semantic tools and functionalities increasing model intelligence (reflection, direct links, semantic computing capabilities), back off automation capabilities, and production of smart tools for PDA, iPhone and a large set of browsers. The Extended

AXMEDIS format includes full multilingual modeling for metadata and descriptors, semantic descriptors in RDF and XML, direct links and access to internal resources, extended behavioral capabilities, semantic computing capabilities, maintaining old packaging and DRM supports.

The paper reported an overview and comparison of the most widespread intelligent content models. The comparison has highlighted the enabling technologies for intelligent content, and it may be used to help the reader in identifying the best model for a given application. The Extended AXMEDIS model and format are supported by a set of tools for content authoring, execution and for massively scalable automated management of content, with the so-called AXCP tools.

It has been shown that with the Extended AXMEDIS model it is possible to produce and distribute intelligent content, and to exploit the legacy content formats such as: HTML, NewsML, MXF, SCORM, ZIP, SMIL, etc. Examples of intelligent content have been presented, while other more complex and complete examples can be recovered on the trial portals which are http://mobmed.axmedis.org and http://xmf.axmedis.org and by downloading the demo tools. Therefore, the usage of the Extended AXMEDIS format to cope with NewsML content is very profitable and interesting, since it allows to exploit innovative functionalities of the intelligent content and therefore it offers advantages with respect to the traditional models.

The experience has allowed us also to identify a small number of topics, which the future intelligent content models should evolve according to, as well as the most meaningful lacks in other very common models and formats. According to the analysis, collaboration and versioning aspects are the most innovative features to be integrated together with semantics aspects. The complexity of managing the versioning, semantics aspects and multiple users and the related authoring tools are the most relevant challenges that have to be solved in the next years in the area of complex intelligent content. The performed analysis has allowed us to draw the design and implementation of the AXMEDIS model and tools. Performing the analysis on the Extended AXMEDIS model has allowed us to realize that among the currently considered models it is one of the most complete and powerful. Additional future work for AXMEDIS is mainly in the direction of enforcing capabilities of collaboration and versioning in the Extended AXMEDIS model and tools. The full documentation can be recovered on the AXMEDIS portal. AXMEDIS is an open platform, which means that the user can join the AXMEDIS community. All the examples mentioned in this paper are accessible on the same web portal.

Acknowledgments

The authors would like to express their thanks to content providers and collaborators of AXMEDIS framework and Mobile Medicine partnership including the Expert User Group and all the affiliated members, for their contribution and collaboration efforts. The authors would also like to express their thanks to the European Commission for partially funding AXMEDIS IP IST FP6 and VARIAZIONI eContentPlus.

References

[1] I. Burnett, R. Van de Walle, K. Hill, J. Bormans, F. Pereira, "*MPEG-21: goals and achievements*", IEEE Multimedia, Vol.10, Issue 4, pp.60-70, Oct-Dec 2003.
[2] I. Burnett, S.J. Davis, G.M. Drury, "*MPEG-21 digital item declaration and Identification-principles and compression*", IEEE Transactions on Multimedia, Vol.7, N.3, pp.400-407, 2005.
[3] AAF/MXF: Multimedia Exchange Format, http://www.ist-nuggets.tv/mxf
[4] NewsML: http://www.iptc.org
[5] M. Kodama, T. Ozono, T. Shintani, Y Aosaki, "*Realizing a News Value Markup Language for News Management Systems Using NewsML*" Complex, Intelligent and Software Intensive Systems, 2008. CISIS 2008. International Conference on 4-7 March 2008 Page(s):249 - 255
[6] Advanced Distributed Learning, "T*he Sharable Content Object Reference Model (SCORM) - Version 1.3 - WD*," March 2003
[7] F. Pereira, T. Ebrahimi, eds. (2002), "*The MPEG-4 Book*" IMSC Press, Prentice Hall.
[8] METS, <http://www.loc.gov/standards/mets/> http://www.loc.gov/standards/mets/METSOverview.v2.html
[9] J. Lee, S.O. Hwang, S.W. Jeong, K.S. Yoon, C.S., Park, J.-C. Ryou, "*A DRM Framework for Distributing Digital Contents Through the Internet*", ETRI Journal, Vol.25, N.6, pp.423-435, December 2003.
[10] P. Nesi, D. Rogai, A. Vallotti, "*A Protection Processor for MPEG-21 Players*". In Multimedia and Expo, 2006 IEEE International Conference. Page 1357—1360. 9-12 July 2006, Toronto, Canada.
[11] MARC: Network development and MARC standards office, library of congress, http://www.loc.gov/marc/ndmso.html
[12] SMIL: Synchronized Multimedia Integration Language, http://www.w3.org/AudioVideo/
[13] MHP: Multimedia Home Platform http://www.mhp.org,
[14] MHEG: Multimedia and Hypermedia information coding Expert Group, ISO/IEC/JTC1SC 29 working group, http://www.mheg.org/users/mheg/index.php
[15] Laser: Lightweight application scene representation, MPEG ISO Laser, http://www.mpeg-laser.org/
[16] SVG: Scalable vector graphic, http://www.w3.org/Graphics/SVG/
[17] Dublin Core Metadata Initiative. Dublin Core Metadata Element Set, Version 1.1. http://dublincore.org/documents/dces/.
[18] Tv-AnyTime Forum: http://www.tv-anytime.org/
[19] OAI, Open Archive Initiative, http://www.openarchives.org/
[20] MPEG-7: http://mpeg.chiariglione.org/standards/mpeg-7/mpeg-7.htm
[21] H.-W. Hilse, J. Kothe, "*Implementing Persistent Identifiers*". Research and Development Department of the Goettingen State and University Library, http://www.knaw.nl/ecpa/publ/pdf/2732.pdf
[22] ACEMEDIA: Integrating knowledge, Semantics and content for user centred intelligent media services, http://www.acemedia.org/aceMedia
[23] S. Bloehdorn, K. Petridis, C. Saathoff, N. Simou, V. Tzouvaras, Y. Avrithis, S. Handschuh, I. Kompatsiaris, S. Staab, and M. G. Strintzis, "*Semantic Annotation of Images and Videos for Multimedia Analysis*", Proc. 2nd European Semantic Web Conference, ESWC 2005, Heraklion, Greece, May 2005.
[24] X-MEDIA: http://www.x-media-project.org

[25] F. Ciravegna, S. Staab & X-Media Consortium. X-Media: Large Scale Knowledge Acquisition, Sharing and Reuse across-Media. LREC'06, 24-26 May, Genoa, Italy.

[26] AXMEDIS, "*Framework and Tools Specifications*", http://www.axmedis.org

[27] P. Bellini, P. Nesi, D. Rogai, *"Exploiting MPEG-21 File Format for cross media conten*t", Proc. of the International Conference on Distributed Multimedia Systems, DMS 2007, September 6-8, 2007, San Francisco Bay, USA, Organized by Knowledge Systems Institute.

[28] P. Bellini, P. Nesi, L. Ortimini, D. Rogai, A. Vallotti, "*Model and usage of a core module for AXMEDIS/MPEG21 content manipulation tools*", IEEE International Conference on Multimedia & Expo (ICME 2006), Pages 577--580 Toronto, Canada, 9-12 July, 2006.

[29] K. Schellner, G.U. Westermann, S. Zillner, W. Klas, "*CULTOS: Towards a World-wide Digital Collection of Exchangeable Units of Multimedia Content for Intertextual Studies*". In Conference of Distributed Multimedia Systems (DMS2003), 2003.

[30] W. Behrendt, A. Gangemi, W. Maass, R. Westenthaler, "*Towards an Ontology-Based Distributed Architecture for Paid Content*". In: A. Gómez-Pérez and J. Euzenat (Eds.): ESWC 2005, LNCS 3532, pp. 257–271, Springer-Verlag Berlin Heidelberg 2005

[31] MPEG Group, "Introducing MPEG-21 DID", http://www.chiariglione.org/mpeg/technologies/mp21-did/index.htm

[32] P. Bellini, P. Nesi, D. Rogai, and A. Vallotti, "*AXMEDIS Tool Core for MPEG-21 Authoring /Playing,*" presented at International Conference on Automating Production of Cross media content for multichannel distribution AXMEDIS2005, Florence, 30 Nov – 2 Dec 2005.

[33] J. Bekaert, P. Hochstenbach, H. Van de Sompel, "*Using MPEG-21 DIDL to Represent Complex Digital Objects in the Los Alamos National Laboratory Digital Library*", D-Lib Magazine, Nov 2003 http://mirrored.ukoln.ac.uk/lis-journals/dlib/dlib/dlib/november03/bekaert/11bekaert.html

[34] CCPP: Composite Capabilities/Preference Profiles, http://www.w3.org/Mobile/CCPP/

[35] MPEG-21 DIA: http://mpeg.chiariglione.org/technologies/mpeg-21/mp21-dia/index.htm

[36] MPEG-21 DIP: http://mpeg.chiariglione.org/technologies/mpeg-21/mp21-dip/index.htm

[37] J. Thiele, "Embedding Spider Monkey - best practice" http://egachine.berlios.de/embedding-sm-best-practice/embedding-sm-best-practice-index.html

[38] MPEG-21 REL: http://mpeg.chiariglione.org/technologies/mpeg-21/mp21-rel/index.htm

[39] OMA DRM: http://www.openmobilealliance.org/technical/release_program/drm_v2_0.aspx

[40] P. Bellini, I. Bruno, P. Nesi, M. Spighi, "*Intelligent Content Model based on MPEG-21*", in Proc. AXMEDIS 2008, Florence, Italy, 17-19 Nov. 2008, pp 41-48, IEEE Press.

[41] P. Bellini, I. Bruno, P. Nesi, ``*A GRID based Framework and tools for automating production of cross media content*'', Proc. of the 11th IEEE International Conference on Engineering of Complex Computer Systems (IEEE ICECCS2006), California, USA, 14-18 August 2006.

[42] P. Bellini, I. Bruno, P. Nesi, ``*A language and architecture for automating multimedia content production on grid*'', Proc. of the IEEE International Conference on Multimedia & Expo (ICME 2006), IEEE Press, Toronto, Canada, 9-12 July, 2006.

[43] P. Bellini, I. Bruno, P. Nesi, "*Visual Programming of Content Processing Grid*", The 15th International Conference on Distributed Multimedia Systems, DMS2009, Hotel Sofitel, Redwood City, San Francisco Bay, USA, 10-12 September, 2009;

[44] A. Vakali and G. Pallis, ''*Content Delivery Networks: Status and Trends,*'' IEEE Internet Computing, vol.7, n.6, pp.68-74, 2003.

[45] D. De Roure, N.R. Jennings, N.R. Shadbolt, "*The Semantic Grid: Past, Present, and Future*", Peoc. of The IEEE, vol.93, n.3, March 2005.

[46] AX4HOME DVB-T Recorder and Broadcast Enhancer, http://www.axmedis.org/documenti/view_documenti.php?doc_id=4005, http://www.axmedis.org/com/index.php?option=com_content&task=view&id=113&Itemid=80