THE MPEG-21 MULTIMEDIA FRAMEWORK: CONVERSIONS AND PERMISSIONS

Christian Timmerer[‡] *· Thomas DeMartini*[†] *· Hermann Hellwagner*[‡]

[‡] Department of Information Technology (ITEC), Klagenfurt University
{christian.timmerer, hermann.hellwagner}@itec.uni-klu.ac.at

[†] ContentGuard Inc., USA
Thomas.DeMartini@CONTENTGUARD.COM

ALPEN-ADRIA
UNIVERSITÄT
KLAGENFURT

Department of Information Technology (ITEC)
Klagenfurt University
Technical Report No. TR/ITEC/06/1.03
March 2006

The MPEG-21 Multimedia Framework: Conversions and Permissions

Christian Timmerer[1] · Thomas DeMartini[2] · Hermann Hellwagner[1]

[1]Department of Information Technology (ITEC), Klagenfurt University
{christian.timmerer, hermann.hellwagner}@itec.uni-klu.ac.at

[2]ContentGuard Inc., USA
Thomas.DeMartini@CONTENTGUARD.COM

Abstract

In the area of multimedia computing and communication, one of the major objectives currently being pursued is Universal Multimedia Access (UMA), *i.e.*, enabling users to transparently access any kind of content from anywhere, anytime, with any device. In the course of the MPEG-21 (Multimedia Framework) activities, the ISO/IEC MPEG standardization group has created essential building blocks toward this goal, mainly metadata standards. Two of them are worth noting here: (1) the Digital Item Adaptation (DIA) specification, including normative vocabulary and formats to describe the multimedia consumption context (device, network, natural environment characteristics, user preferences) and to steer media adaptation operations required for UMA; (2) the Rights Expression Language (REL) and Rights Data Dictionary (RDD) specifications, providing terms and a language to express permissions on the usage of the media (who, how, what, under which conditions).
In this paper, we describe a recent MPEG-21 DIA Amendment in this area, specifying description formats for: (1) *multimedia conversion capabilities*, which tool or service providers may use to normatively specify the media adaptation (conversion) capabilities of their tools or services, respectively; (2) *permissions and conditions for multimedia conversions*, which can be utilized by content providers to determine which adaptations (changes) are permitted on their contents under what conditions. The latter description format embeds media adaptation descriptions into rights expressions, filling a gap between DIA and REL/RDD. In the paper, a use case illustrating a complex UMA scenario justifies the need for these descriptions. Exemplary conversions and permissions descriptions that apply to this use case as well as detailed explanations will be given in the main portion of the paper.

1 Introduction

In future pervasive and ubiquitous media environments, users are envisaged to access and interact with different types of multimedia content independent of their access devices and access networks; the users' preferences and the natural environment where the content will be consumed, will be taken into account as well. Research issues resulting from these requirements are generally referred to as Universal Multimedia Access (UMA) [1]; *i.e.*, UMA aims to provide mechanisms for accessing any kind of content from anywhere, anytime, with any device type. Some answers to the above issues have been considered within standardization bodies such as the ISO/IEC Moving Picture Experts Group (MPEG) which is well known by their audio-visual coding standards (MPEG-1/-2/-4) and also multimedia metadata standards

(MPEG-7). In practice, however, after MPEG-7 was near to its completion, it has been recognized that – although many building blocks enabling UMA are in place – no big picture of how these tools work together exists. As a consequence, MPEG launched a new work item, referred to as the MPEG-21 Multimedia Framework [2], which aims to provide the big picture for a secure and interoperable exchange of multimedia resources across heterogeneous devices, networks, and users.

MPEG-21 provides facilities for interoperable transactions of Digital Items among Users, taking into account the devices' and networks' heterogeneity. (Digital Items in MPEG-21 are comprised of multimedia resources and metadata within a standardized structure. Users with a capitalized "U" in MPEG-21 include individuals, communities, organizations, and governments.) More specifically, issues resulting from this heterogeneity are addressed by Part 7 of MPEG-21 which is known as Digital Item Adaptation (DIA) [3].

At the same time, digital rights management (DRM) issues are becoming more and more important in today's multimedia infrastructures; they are addressed by Parts 5 and 6 of MPEG-21, *i.e.*, the Rights Expression Language (REL) and the Rights Data Dictionary (RDD), respectively [4].

In this paper, we present an interoperable approach enabling fine-grained control (*e.g.*, expressing permissions) over the changes that can occur when playing, modifying, or adapting multimedia contents (embedded in Digital Items), based on metadata as specified within MPEG-21 REL, RDD, and DIA. The remainder of this paper is organized as follows. Section 2 provides a brief overview of MPEG-21 DIA and REL/RDD. Section 3 presents an illustrative use case for which permissible adaptation of multimedia resources is becoming self-evident. The actual descriptions enabling the development of permissible and interoperable adaptation services are detailed in Section 4. Section 5 concludes the paper.

2 Background

2.1 MPEG-21 Digital Item Adaptation

As briefly outlined in the previous section, MPEG-21 Digital Item Adaptation (DIA) addresses today's heterogeneity of devices and networks by specifying normative description formats (or description "tools", in MPEG-21 terminology) providing assistance in the adaptation of Digital Items. In particular, the DIA standard specifies means enabling the construction of device- and coding format-independent adaptation engines. Device independence is achieved through a standardized vocabulary and model expressed using XML Schema in order to describe the usage environment with regard to the characteristics of the users, the capabilities and conditions of networks and devices as well as the natural environment in which Digital Items are consumed. Coding format independence is referred to as the possibility to facilitate a metadata-driven and coding format-agnostic adaptation engine for all kinds of (scalable) multimedia bit-streams, *e.g.*, MPEG-4 video/audio and JPEG2000.

The high-level architecture of DIA is depicted in Figure 1. As shown in the figure, only tools used to guide the adaptation engine are specified by DIA, the adaptation engines themselves are left open to industry competition.

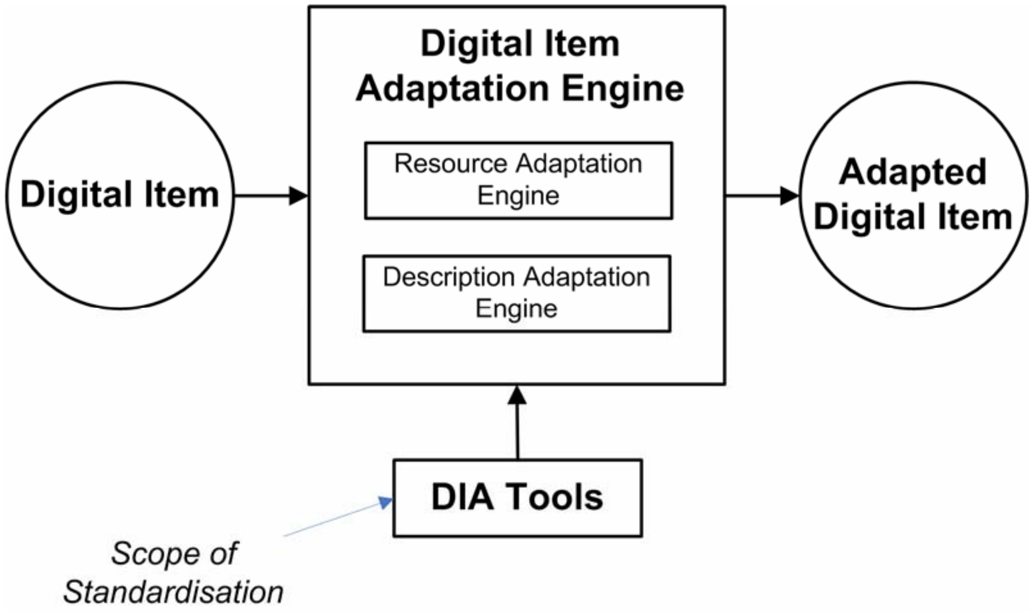

Figure 1 — Concept of MPEG-21 DIA [3].

The first amendment to DIA [5] provides the description of fine-grained media conversions by means of the conversion operations' names and parameters, which can be used to define rights expressions to govern adaptations in an interoperable way. Before going into detail how this can be achieved, we give a brief overview of how rights can be expressed by means of the MPEG-21 REL and RDD. Note that the latter is tightly coupled with the REL.

2.2 MPEG-21 Rights Expression Language and Rights Data Dictionary

The REL specification establishes a well-structured, flexible language for the unambiguous and machine-interpretable expression of permissions. An expression written in this language is called a rights expression or license.

Licenses are the fundamental unit of communication in the rights domain. They are generated by license issuers and communicated to other system entities. When one of these system entities, *p*, wishes to perform action *a* on (multimedia) resource *r*, it uses its licenses, *L*, as part of an authorization request asking the question: "According to licenses *L* and given assumptions *R*, is *p* permitted to do action *a* to resource *r* during time interval *v* in context *x*?" If the answer is "Yes", then *p* proceeds with the action; otherwise, *p* does not.

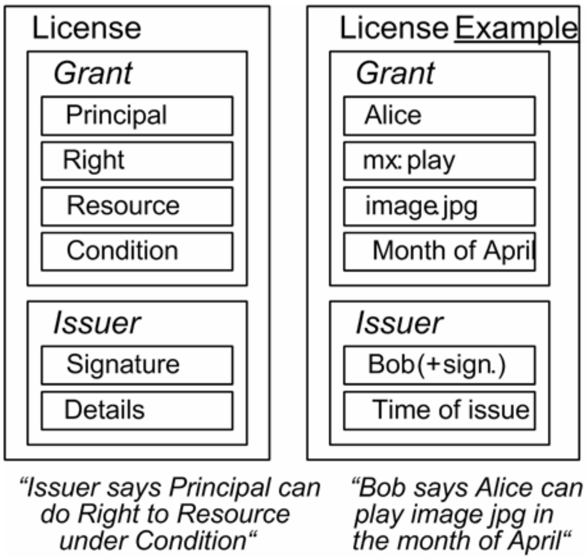

Figure 2 —REL data model [4].

Licenses are structured to contain all the information necessary to answer such requests. Using advanced license structures, it is possible for a single license to authorize many different system entities to do many different actions on many different resources under many different conditions. A simple-structured license, as depicted in Figure 2 (*License*), has two key parts: a set of grants and a field identifying the issuer (*Issuer*) of those grants. A simple-structured grant (*Grant*) has fields for identifying, in order, a system entity (the *Principal* field), an action (the *Right* field), a resource (the *Resource* field), and some conditions (the *Condition* field).

The system entity and resource in the grant (and the issuer in the license) are identified using technology from XML Digital Signature [6]. The conditions (time, fee, count, territory, freshness, integrity, marking, signed-by, and so forth) are specified using a variety of technologies ranging from `xsd:DateTime` to WSDL [7] and UDDI [8] to XML Digital Signature. The actions are identified using terminology from the RDD.

The RDD provides a set of clear, consistent, structured, integrated, and uniquely identified terms. These terms form the framework for an extensible dictionary of action terms with clear relationships to one another. Fourteen of these action terms have shorthand representations in the REL, *e.g.*, `mx:play` which identifies the act of deriving a (possibly-adapted) transient and directly perceivable representation of a resource. The other action terms in the RDD (beyond the fourteen) as well as any new terms that might be registered with the RDD registration authority over time can be represented in the REL with the longhand URI notation. By resolving such a URI with an RDD implementation, it is possible to determine how the corresponding action term is related to other terms in the dictionary. For example, one action term might be a specialization or generalization of another action term.

3 Use Case

Consider the case – illustrated in Figure 3 – of a streaming server that serves content from company A and company B to a variety of devices, such as a personal computer (PC), television (TV), or mobile device. Each of these devices might have different display capabilities. For example, the PC might have the highest spatial resolution, while the TV and mobile device have lower spatial resolutions. On the other hand, the TV might be a monochrome TV, while the PC and mobile device might have color displays. As mentioned above, the DIA specification provides the technologies for these various devices to communicate their display capabilities to the streaming server in an interoperable fashion. Utilizing this information, the streaming server needs to come up with an adaptation plan to provide the optimum content to each device. Typically, it would be very difficult to build a monolithic streaming server module that was able to do any kind of transformation itself. Instead, the streaming server would probably make use of a number of smaller adaptation engines installed on the server (or located somewhere else within the delivery chain), such as a greyscaler, a spatial scaler, or an image/video cropper.

In order for the streaming server to take advantage of new adaptation engines as they are installed, *there needs to be an interoperable way for the adaptation engines to express their conversion capabilities to the streaming server*.

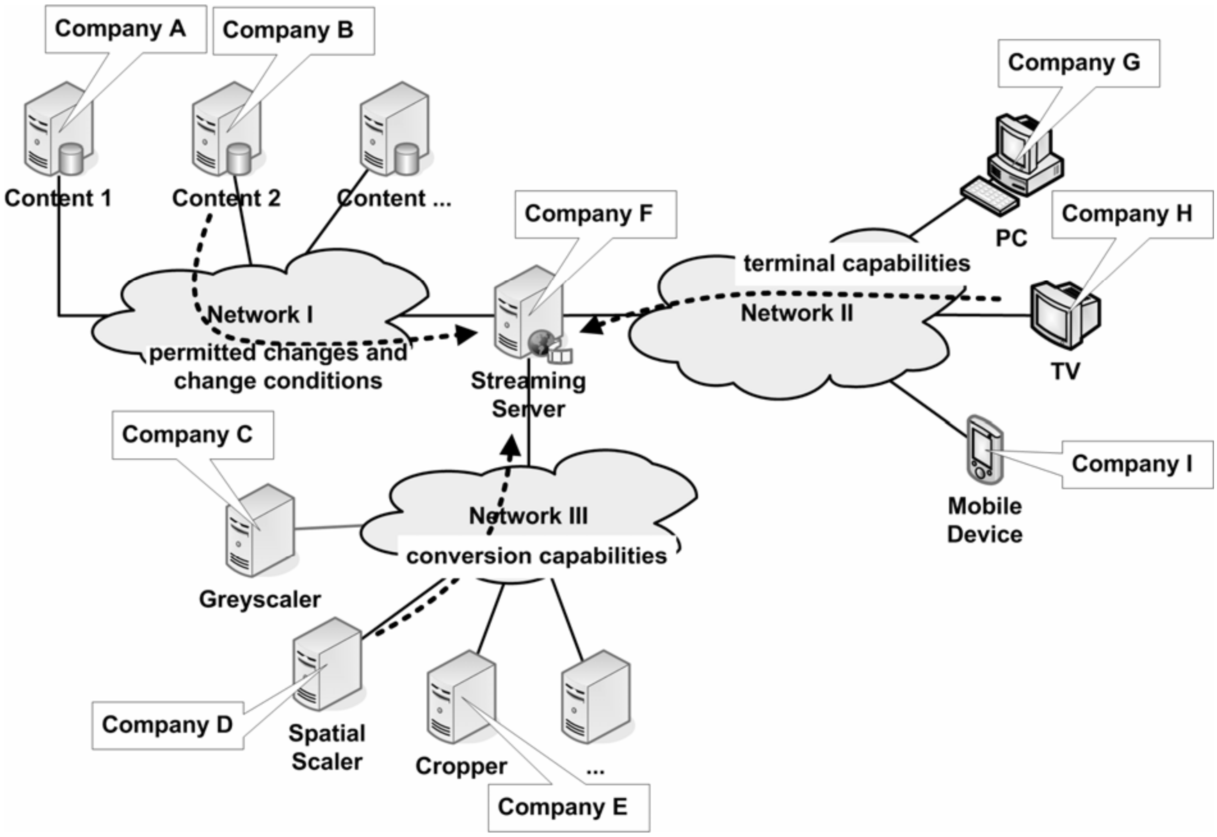

Figure 3 — Use case for permissible Digital Item Adaptation.

However, it is also important that the adaptation process does not get out of control when more and more adaptation engines are added. The first step for controlling adaptations is for company F to have some control over which adaptation engines get used by its streaming server. One way it can do this is to check each adaptation engine before installing it on its streaming server. Another way is for companies C, D, and E to sign their adaptation engines and for the streaming server software to verify the signature as belonging to a trustworthy company before using an adaptation engine.

The next step for controlling adaptations – from the content owners' points of view – is for the streaming server to understand (and respect) which adaptations are permitted to be performed for each customer. Content owners have a legitimate interest in controlling which adaptations of their content are permitted. For example, a color-reduced version of a computer generated animated film might look very bad and result in an adverse viewer opinion that gets shared with the viewer's friends and diminishes the sales of the film. In such a case, the color-reduced version might not be available until much later in the release cycle. In another example, a reduced-resolution version of a movie might be available at a significant discount, whereas the high resolution version can only be viewed by those paying the regular price. In order for the streaming server to understand which adaptations are permitted to be performed for each customer according to the content owners' interests, *there needs to be an interoperable way for the content owners to express the consumers' permissions to the streaming server*.

4 Permissible Digital Item Adaptation

The first amendment to MPEG-21 Digital Item Adaptation [5] provides appropriate answers to the issues raised in the previous section. In particular, it provides normative description tools – among others – enabling the description of conversions a terminal is capable of performing. Additionally, it provides means for flawless integration of these descriptions within licenses for DRM-related purposes. The former description formats are referred to as *conversion capabilities* and are provided by the companies that are implementing the actual conversion tools (*e.g.*, provided by the companies C, D, and E of Figure 3). The latter description formats fall under the umbrella of *permitted DIA changes* and *change conditions*, respectively, which are actually used by content providers in order to determine which conversions are permitted under what kind of conditions (*e.g.*, provided by the companies A and B of Figure 3). Thereafter, the conversion capabilities are used to discover and instantiate (*e.g.*, accomplished by company F of Figure 3) the conversion tools according to the usage environment description (*e.g.*, provided by the devices of companies G, H, and I of Figure 3).

4.1 Conversion Capabilities

The conversion capabilities are designed in a generic way allowing for easy integration of existing description formats provided by existing MPEG standards as well as other communities or standards coming from other standardization bodies. In particular, the conversion capabilities are part of the DIA terminal capabilities comprising of a reference to a possibly specialized RDD term followed by any kind of XML-based description.

As an example, the Semantic Web Services community provides means for describing such kind of capabilities as a semantic, knowledge-based extension to the well-known Web Ontology Language (OWL) [9], also known as OWL-S [10]. This approach is based on the IOPE (input, output, pre-conditions, and effects) concept known from Artificial Intelligence planning techniques [11] which is also applicable for multimedia multi-step adaptations [16] required for the use case as presented in the previous section. In particular, OWL-S not only describes what the service actually does but provides also a grounding mechanism enabling an interoperable instantiation of the service during runtime. However, only the former is primarily used for DRM purposes, *i.e.*, the selection of the available conversion operations identified by their conversion capabilities description against the permitted DIA changes taking into account several change conditions, as exemplified in Document 1. Note that such kind of descriptions are used by companies providing the actual implementations of the conversion tools such as highlighted in Section 3 (*i.e.*, companies C, D and E).

```
<DIA>
 <Description xsi:type="TerminalsType">
  <Terminal>
   <TerminalCapability xsi:type="ConversionCapabilitiesType">
    <ConversionCapability>
     <ConversionActUri
          uri="urn:mpeg:mpeg21:2003:01-RDD-NS:CropRectangularBitmapImage"/>
     <rdf:RDF>
      <!-- import of the OWL-S ontology, i.e., profile, process,
           the actual cropping process, and the grounding -->
      <hierarchy:CropRectangularBitmapImage rdf:ID="CropImage">
       <!-- reference to the process model specification -->
       <profile:has_process rdf:resource="&crop_process;#CropImage"/>
       <profile:serviceName>CropImage</profile:serviceName>
       <!-- description of inputs -->
       <profile:hasInput rdf:resource="&crop_process;#roi-imageIn"/>
       <profile:hasInput rdf:resource="&crop_process;#roi-width"/>
       <profile:hasInput rdf:resource="&crop_process;#roi-height"/>
       <profile:hasInput rdf:resource="&crop_process;#roi-x"/>
       <profile:hasInput rdf:resource="&crop_process;#roi-y"/>
       <!-- description of outputs -->
       <profile:hasOutput rdf:resource="&crop_process;#imageOut"/>
       <!-- description of effects -->
       <profile:hasEffect rdf:resource="&crop_process;#eff_resolution"/>
      </hierarchy:CropRectangularBitmapImage>
     </rdf:RDF>
    </ConversionCapability>
   </TerminalCapability>
  </Terminal>
 </Description>
</DIA>
```

Document 1 — Conversion Capabilities example describing an image cropper.

This example shows an OWL-S description (*i.e.*, only the service profile) for an image cropper embedded within an MPEG-21 DIA conversion capabilities description identified by the conversion act `CropRectangularBitmapImage` which is a specialization of the RDD term "adapt". The service profile provides a concise description of the service to a registry, but once the service has been selected the service profile is useless; rather, the entity making use of the service will use the service model (not shown here due to space restrictions) to control the interaction with the service. Finally, the service grounding (*i.e.*, the third part of an OWL-S description and also not shown here due to space restrictions) specifies the details of how an agent can access a service. For further details about the service model and grounding mechanism the interested reader is referred to Annex I of [5].

However, the essence of a service profile is the specification of what functionality the service provides and the specification of the conditions that must be satisfied for a successful result. The OWL-S service profile represents two aspects of the functionality of the service. First, the information regarding the transformation (*i.e.*, represented by inputs and outputs) and second, the state change produced by the execution of the service (*i.e.*, represented by preconditions and effects). For example, a cropping service – as shown in Document 1 – for images may require the original image as input as well as the desired region of interest, *i.e.*, width, height, and x/y-offset of the image to be cropped. The output would be an image, and the effect of applying the service is a change in resolution.

Due to the large number of possibilities for describing conversion capabilities – others include WSDL-S [12], SWSL [13], or WSMO [14] – the current version of the DIA standard deliberately excludes a normative reference to one of them. The future will show which approach will be adopted, *e.g.*, by the World Wide Web Consortium, and possibly included within another amendment of MPEG-21 Digital Item Adaptation.

4.2 Permitted DIA Changes and Change Conditions

The same reference to possibly specialized RDD terms used in terminal capabilities are also used in the licenses defining which changes (*i.e.*, permitted DIA changes) are allowed under which conditions (*i.e.*, change conditions). Both are specified as authorization context properties within the DIA standard. The former, *i.e.*, permitted DIA changes, identify the conversion being used for the formal or format changes (according to the semantics of the right member of the authorization request) during the requested performance. The latter, *i.e.*, change conditions, allow for defining constraints that must be satisfied, *i.e.*, evaluated to true, according to the permitted DIA changes. In particular, the constraints are formulated using an XML-based version of the well-known reverse polish notation (RPN) [15].

Document 2 exemplifies permitted DIA changes and change constraints embedded within an REL license, *i.e.*, as a child of the `r:allConditions` element. The permitted DIA changes comprise of a list of conversion acts which are most likely references to possibly specialized RDD terms (in this case, the reference is to the `CropRectangularBitmapImage` term). Note that all other conversions are by default not permitted. In some cases, a reference to a very generic RDD term such as "adapt" would allow many kinds of adaptations according to the definition of the term "adapt" within the RDD.

```
<r:license>
 <r:inventory>
  <!-- ... -->
 </r:inventory>
 <r:grant>
  <!-- Barney may play the video ... -->
  <!-- Principal: Barney -->
  <r:keyHolder licensePartIdRef="Barney"/>
  <!-- Right: play -->
  <mx:play/>
  <!-- Resource: the video -->
  <mx:diReference licensePartIdRef="video"/>

  <!-- ... under these Conditions -->
  <r:allConditions>
   <dia:permittedDiaChanges>
    <dia:ConversionDescription xsi:type="dia:ConversionUriType">
     <dia:ConversionActUri
       uri="urn:mpeg:mpeg21:2003:01-RDD-NS:CropRectangularBitmapImage"/>
    </dia:ConversionDescription>
    <!-- further permitted DIA changes would go here -->
   </dia:permittedDiaChanges>

   <!-- these constraints apply whether or not the image is cropped -->
   <dia:changeConstraint>
```

```
    <dia:constraint>
     <dia:AdaptationUnitConstraints>

      <!-- width must be less than 352 -->
      <dia:LimitConstraint>
       <dia:Argument xsi:type="dia:SemanticalRefType"
         semantics="urn:mpeg:mpeg21:2003:01-DIA-MediaInformationCS-NS:17"/>
       <dia:Argument xsi:type="dia:ConstantDataType">
        <dia:Constant xsi:type="dia:IntegerType">
         <dia:Value>352</dia:Value>
        </dia:Constant>
       </dia:Argument>
       <dia:Operation
         operator="urn:mpeg:mpeg21:2003:01-DIA-StackFunctionOperatorCS-NS:12"/>
      </dia:LimitConstraint>

      <!-- height must be less than 240 -->
      <dia:LimitConstraint>
       <dia:Argument xsi:type="dia:SemanticalRefType"
         semantics="urn:mpeg:mpeg21:2003:01-DIA-MediaInformationCS-NS:18"/>
       <dia:Argument xsi:type="dia:ConstantDataType">
        <dia:Constant xsi:type="dia:IntegerType">
         <dia:Value>240</dia:Value>
        </dia:Constant>
       </dia:Argument>
       <dia:Operation
         operator="urn:mpeg:mpeg21:2003:01-DIA-StackFunctionOperatorCS-NS:12"/>
      </dia:LimitConstraint>

      <!-- final aspect ratio must equal input aspect ratio -->
      <dia:LimitConstraint>
       <dia:Argument xsi:type="dia:SemanticalRefType"
         semantics="urn:mpeg:mpeg21:2003:01-DIA-MediaInformationCS-NS:19"/>
       <dia:Argument xsi:type="dia:SemanticalDataRefType"
         semantics="urn:mpeg:mpeg21:2003:01-DIA-MediaInformationCS-NS:19"/>
       <dia:Operation
         operator="urn:mpeg:mpeg21:2003:01-DIA-StackFunctionOperatorCS-NS:11"/>
      </dia:LimitConstraint>

     </dia:AdaptationUnitConstraints>
    </dia:constraint>
   </dia:changeConstraint>
  </r:allConditions>
 </r:grant>
 <r:issuer>
  <r:keyHolder licensePartIdRef="Anthony"/>
 </r:issuer>
</r:license>
```

Document 2 — REL license example including permitted DIA changes with change conditions.

In either case, the permitted DIA changes could be further restricted by using the `dia:changeConstraint` element which usually appears as a sibling of the `dia:permittedDIAChanges` element. In the above example (Document 2) three change constraints are defined.

The first constraint defines that the width of the resulting image must be smaller than 352 pixels. The corresponding RPN would look like as follows: 'width$_{img}$', '352', '<'. The attribute to test (in this case image width) is denoted by a reference (`...MediaInformationCS-NS:17`) to a term defined within the media information classification scheme and the operator (in this case the less-than operator) is denoted by a reference (`...StackFunctionOperatorCS-NS:12`) to a term defined within another classification scheme named stack function operator. A classification scheme is a hierarchically arranged extensible list of terms with corresponding semantics written in natural English language. As such, it is similar to an ontology except without means for automatically reasoning among the terms defined in the classification scheme.

The second constraint defines a similar constraint as the first constraint but for the height of the image. The third constraint defines that the resulting aspect ratio must be equal to the input aspect ratio.

Note that the licenses are usually provided by the content owners (*i.e.*, companies A and B of the use case described in Section 3). Finally, the actual service provider (*i.e.*, company F) needs to enforce the rights within the licenses of the content owners and instantiate the conversion tools in an appropriate way. A possible way of achieving a correct instantiation of conversion tools is described in [16].

In this particular case, the license says Anthony allows Barney to play the video in its original form (assuming the original form is already smaller than 240x352) or cropped to any degree necessary to fit on his mobile device's screen, maintain the aspect ratio, and be no larger than 240x352. When Barney's device connects to the streaming server via Network II of Figure 3, it informs the streaming server that its maximum resolution is, say, 240x320. Suppose the original video is 264x352. The streaming server evaluates the license and determines that Barney is permitted to receive the video at 240x320 (since 240x320 is smaller than 240x352 and of the same aspect ratio as 264x352). The question now is if the streaming server can adapt the video to that resolution using only the "cropping" technique (since that's the only conversion allowed). To answer this question, the streaming server looks at the terminal capabilities of its installed conversion tools, finds the ones that can do cropping, loads them with the correct parameters, generates the cropped result, and streams the result to Barney's device.

5 Conclusion

In this paper we have presented means for describing conversion capabilities of certain terminals as well as means for describing permitted changes and change constraints enabling interoperability among conversion tool and content providers. These new description formats bridge the gap between the tools as defined within Digital Item Adaptation (part 7 of MPEG-21) and the Rights Expression Language (part 5 of MPEG-21). Furthermore, the consideration of digital rights management issues regarding the adaptation of digital media content is yet another important step towards enabling the aim of Universal Multimedia Access.

The proposed description formats have been evaluated on a vital use case with appropriate description examples which are applicable to be used within an MPEG-21-compliant DRM-aware environment. Nonetheless, MPEG-21 deliberately does not provide specifications for a complete DRM system but provides components for Intellectual Property Management and Protection (IPMP) within part 4 [17] of the multimedia framework. These components enable the construction of DRM systems including the signaling of the cryptographic mechanisms used for en-/decrypting Digital Items or parts thereof. However, the integration of all these components is beyond the scope of this paper but could be part of any future work.

References

[1] A. Vetro, C. Christopoulos, and T. Ebrahimi (eds.), *IEEE Signal Processing Magazine, Special Issue on Universal Multimedia Access*, vol. 20, no. 2, March 2003.

[2] F. Pereira, J. R. Smith, and A. Vetro (eds.), *IEEE Transactions on Multimedia, Special Section on MPEG-21*, vol. 7, no. 3, June 2005.

[3] A. Vetro and C. Timmerer, "Digital Item Adaptation: Overview of Standardization and Research Activities", *IEEE Transactions on Multimedia*, vol. 7, no. 3, June 2005, pp. 418-426.

[4] X. Wang, T. DeMartini, B. Wragg, M. Paramasivam, and C. Barlas, "The MPEG-21 Rights Expression Language and Rights Data Dictionary", *IEEE Transactions on Multimedia*, vol. 7, no. 3, June 2005, pp. 408-417.

[5] C. Timmerer, T. DeMartini, and Chris Barlas (eds.), "Information technology — Multimedia framework (MPEG-21) — Part 7: Digital Item Adaptation, AMENDMENT 1: DIA Conversions and Permissions", *ISO/IEC 21000-7 AMD/1*, Final Draft Amendment 1 (FDAM/1), October 2005.

[6] D. Eastlake, J. Reagle, and D. Solo (eds.), "XML-Signature Syntax and Processing", *W3C Recommendation*, February 2002. Available at http://www.w3.org/TR/xmldsig-core/.

[7] E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana, "Web Services Description Language (WSDL) 1.1", *W3C Note*, March 2001. Available at http://www.w3.org/TR/2001/NOTE-wsdl-20010315.

[8] L. Clement, A. Hately, C. von Riegen, and T. Rogers (eds.), "UDDI Version 3.0.2", *OASIS UDDI Specification Technical Committee Draft*, Oct. 2004. Available at http://uddi.org/pubs/uddi_v3.htm.

[9] D. L. McGuinness and F. van Harmelen (eds.), "OWL Web Ontology Language", *W3C Recommendation*, February 2004. Available at http://www.w3.org/2004/OWL/.

[10] D. Martin *et al.*, "Bringing Semantics to Web Services: The OWL-S Approach", *Proceedings of the First International Workshop on Semantic Web Services and Web Process Composition (SWSWPC 2004)*, San Diego, California, USA, July, 2004.

[11] M. Fox and D. Long, "PDDL2.1: An Extension to PDDL for Expressing Temporal Planning Domains", *Journal on Artificial Intelligence Research*, vol. 20, 2003, pp. 61–124.

[12] R. Akkiraju, J. Farell, J.A. Miller, M. Nagarajan, A. Sheth, and K. Verma, "Web Service Semantics – WSDL-S", W3C Member Submission, April 2005. Available at http://www.w3.org/2005/04/FSWS/Submissions/17/WSDL-S.htm.

[13] S. Battle *et al.*, "Semantic Web Services Framework (SWSF) Version 1.0", *Semantic Web Services Language (SWSL) Committee*, May 2005. Available at http://www.daml.org/services/swsf/1.0/.

[14] D. Roman, H. Lausen, and U. Keller (eds.), "D2v1.2. Web Service Modeling Ontology (WSMO)", *WSMO Final Draft*, April 2005. Available at http://www.wsmo.org/TR/d2/v1.2/.

[15] C. L. Hamblin, "Computer Languages", *The Australian Journal of Science*, vol. 20, 1957, pp. 135-139. Reprinted in *The Australian Computer Journal*, vol. 17, no. 4, November 1985, pp. 195-198.

[16] D. Jannach, K. Leopold, C. Timmerer, and H. Hellwagner, "A Knowledge Based Approach for Multi-step Media Adaptation", *Proceedings of the 5th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)*, Instituto Superior Tecnico, Lisboa, Portugal, April 2004.

[17] S. Watt, Z. Huang, E. Rodriguez, S. Lauf (eds.), "Information technology — Multimedia framework (MPEG-21) — Part 4: IPMP Components", *ISO/IEC 21000-4*, Final Draft International Standard, October 2005.