Web-based Controlled Delivery and Consumption of Multimedia in MPEG-21 Framework

Hendry1), Kyung-Ae Cha1), Keunsoo Park2) and Munchurl Kim1)

Multimedia Computing & Communications Broadcasting Laboratory (MCCB-Lab) [1]
Information & Communications University (ICU)
119 Munji Street, Yuseong-gu, Daejeon, 305-714, Korea
{hendry, chaka, mkim}@icu.ac.kr

Technical Research Institute2)
Korean Broadcasting System (KBS)
18 Yoido-dong Youngdungpo-gu 150-790 Seoul, Korea
kspark@kbs.co.kr

Abstract

MPEG-21 is an on-progress project of the Moving Picture Expert Group (MPEG) that has been started in many independent parts such as part for standardizing the declaration of a digital item (MPEG-21 Part 2: Digital Item Declaration), right expression language (MPEG-21 Part 5: Rights Expression Language), way to process the digital item (MPEG-21 Part 10: Digital Item Processing), etc. All this part is intended to form a framework to enable transparent and augmented use of multimedia resources across a wide range of networks and devices [1]. In this paper, we show our research initiative to develop a web-based application that utilized many parts of MPEG-21 technology. The implemented concept uses a digital item and license providers in the server side while the MPEG-21 Conformant application is implemented as a plug-in application that will be downloaded to the user terminal when user accesses the webpage for the first time. The result of this research shows that with some combination of MPEG-21 parts, we can develop an authorized multimedia delivery and consumption application.

1. Introduction

According to its vision document, the aim of MPEG-21 framework is "to enable transparent and augmented use of multimedia resources across a wide range of networks and devices, specifically taking into account Intellectual Property Management and Protection and the heterogeneity of the access and delivery infrastructure" [1]. To achieve this vision, MPEG-21 project has been started in many independent parts such as part for standardizing the declaration of a digital item (MPEG-21 Part 2: Digital Item Declaration), right expression language (MPEG-21 Part 5: Rights Expression Language), way to process the digital item (MPEG-21 Part 10: Digital Item Processing), etc. However, there is no "big picture" to describe all these parts glued together to form a multimedia delivery and consumption yet.

Having no "big picture"can be good and bad as well. It is good since all parts of MPEG-21 can be independent so that developer who wants to utilize MPEG-21 technology can freely use parts that he needs and develops his own architecture. However, it is difficult to see how each part interrelated to each other since there is no description how to glue them together. In this paper, we show our research initiative to develop a web-based application for multimedia delivery and consumption that utilizes many parts of MPEG-21 technology.

Our application is developed based on some parts of MPEG-21. We utilized MPEG-21 Digital Item Declaration (DID) to model the digital item to be delivered to the user. The processing of the DID is then controlled by the method defined by MPEG-21 Digital Item Processing (DIP). Finally, to ensure authorized multimedia consumption, the consumption of the resource in the user terminal is controlled by the license defined by using MPEG-21 Rights Expression Language (REL).

The organization of this paper is as follow. In section two, we present our test bed application framework together with background support by MPEG-21 parts. In section three, we present our experimental result. Finally in section four, we conclude our work together with future work.

2. Application Framework

2.1. Test Bed Architecture

Figure 1 shows the architecture of our test bed, where the DID, REL and DIP tools are integrated. The end users connect to the Digital Item Provider to get their expected media contents. Here the media contents are formed as Digital Item (DI) which conveys the resources as well as the user's corresponding license. Upon receiving the request from the terminal, the Digital Item Provider prepares the DI and request licenses appropriate for the DI contents according to the user/terminal from the License Generator/Provider. The process of user/terminal identification can be done in various ways, for example by requiring the user to log-in first before getting the DI. Having the DI and licenses information, the Digital Item Provider combines them together and then sends it to the terminal.

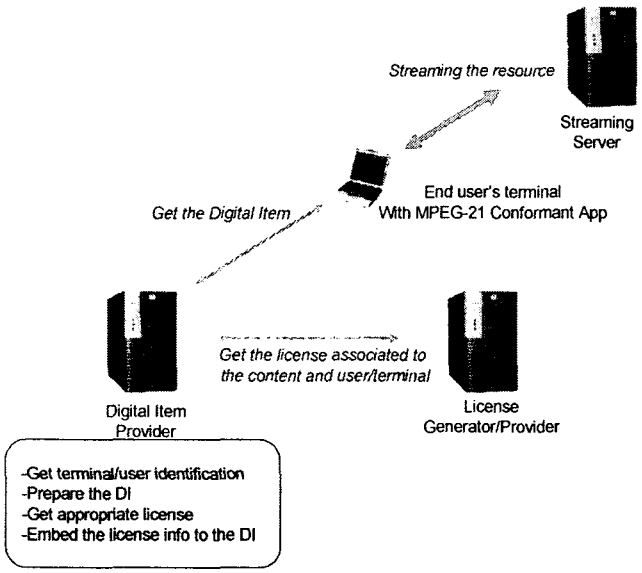

Figure 1. Test bed architecture.

At the terminal side, upon obtaining the DI, the MPEG-21 conformant application parses the DI and presents the item(s) to the user as described by the Digital Item Method which embedded in the DI so that the user consumes the resources according to the license.

In the next subsection, we give more explanation about parts of MPEG-21 technology that we use in our application.

2.2. MPEG-21 Digital Item Declaration (DID)

The media contents which will be consumed by the end users are encapsulated in the form of DID. DID models the Digital Item as the uniform, flexible, and interoperable schema [1]. It is an XML which structure is designed in hierarchical way as shown in the figure 2.

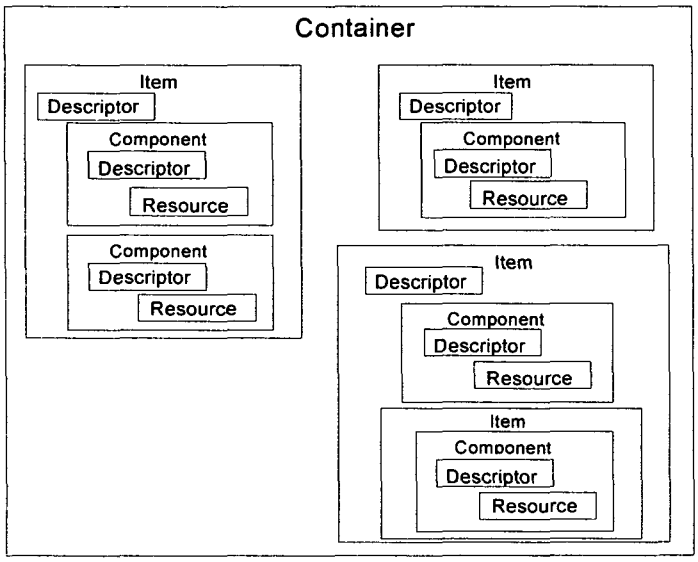

Figure 2. Important elements in DID model [1].

In brief, there is a root "Container" that encapsulates "Item". Each "Item" can have "Component" as the container of "Resource". To give description of each element, a "Descriptor" element may be attached.

2.3. MPEG-21 Rights Expression Language (REL)

In MPEG-21 Vision, Technology, and Strategy document, REL is described as "a machine-readable language that can declare rights and permission regarding the usage of DIs. It is intended to provide flexible, interoperable mechanisms to support transparent and augmented used of digital resources in publishing, distributing, and consuming of digital content" [1].

MPEG-21 REL is modeled by using XML. Its schema contains four basic elements such as:

- The principal that encapsulates the identification of principals to whom a rights are granted.
- The right that the grant specifies to exercise against resource under some condition.
- The resource to which the right in the grant applies.
- The condition that must be met before the right can be exercised.

Figure 3 shows the MPEG-21 REL data model that comprises those four elements above. The data model defines a right associated with a specific resource and issued to a principal. To exercise this right, a set of condition is applied.

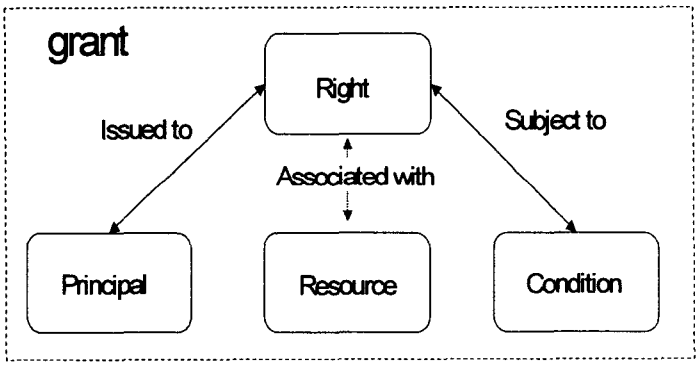

Figure 3. MPEG-21 REL data model [1]

2.4. MPEG-21 Digital Item Processing (DID)

DIP defines a mechanism on how to process a Digital Item. The heart of the DIP lies on the concept of Digital Item Method (DIM). In MPEG-21 Vision, Technology, and Strategy document, the concept of DIM is described as "on receipt of a Digital Item, a user will have available a 'list' of processes ('methods') that can be applied to the Digital Item. These methods provide a mechanism for a User (author, publisher, distributor, etc) to specify a preferred set of procedures by which the DI should be handled." [1]. It is a tool for expressing the intended interaction of a User with a Digital Item (DI) at the level of DID. User will be able to create methods for their Digital Items by using a standardized syntax, DIM, which is based on the third edition of ECMAScript. MPEG intends the DIM to be run on a method engine (called the Digital Item Method Engine, DIME) [1][4].

The DIM will be executed by DIME at a Peer device/application. For interoperability, on its execution time, a DIM will run a the basis on a base set of normative functionality, which in DIP is called "Digital Item Base Operation" or DIBO, that expected to be provide by a Peer [4]. Figure 4 shows the model of the DIP. Upon receiving a DI (declared in DID) which contains DIMs; the DIME will present a list of DIMs to the end user. All the low level accesses to the terminal/peer environment are handled by a normative set of DIBOs.

In our test bed, the DIMs represent the usage ways of the resources. Moreover they have a role to guide users in appropriate consumptions for the resource.

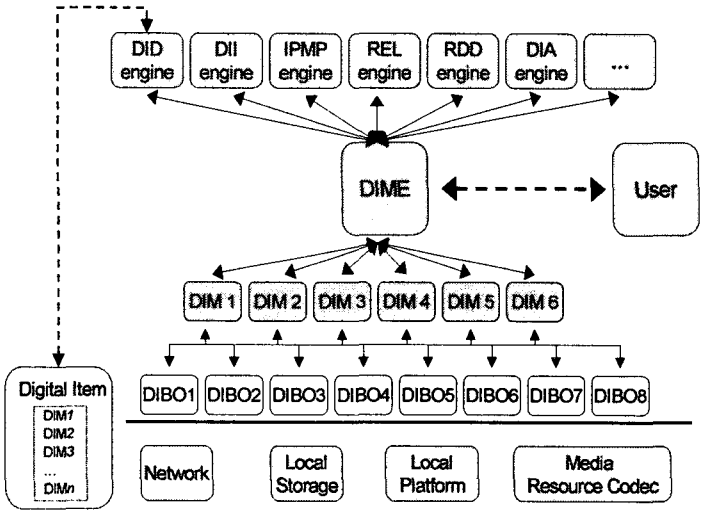

Figure 4. MPEG-21 DIP model [4]

2.5. The integration of DID, REL, and DIP for authorized processing of DI

We have noticed that DID and DIP have been strongly coupled by the concept of DIM. Thus, to achieve a linkage among those three parts we need to define a mechanism how REL be able to relate to DID and DIP. In our experiment, we use approach such as:

- We embed the expression of rights and permission regarding the usage of Digital Item in the Descriptor element of the DID.
- We use and propose a set of DIBOs that will reflect the common practice of REL so that Users (the DIM authors) can put action(s) to exercise the license/permission before the content is consumed by the end user. For example, prior to render a video, the DIM need to see whether there is a permission to do so or not. We call those DIBO as REL-DIBO.

We have analyzed the REL specification and reference software in order to see an efficient set of REL-DIBO interfaces for DIP which results as follow:

- GetRightsInformation DIBO. This DIBO is used to get the license information for the utilization of a certain resource.
- AuthenticatePrincipal DIBO. Upon receiving license information, this DIBO authenticates whether the current user of the terminal is the person who receives the license or not.
- ValidateRightsCondition DIBO. This DIBO validates the conditions imposed in a license. A violation to one of conditions may cause this DIBO return false output and consumption process of a resource should be stopped.

3. Experimental Result

We implement the concept in the form of web application. The digital item provider and license provider are implemented in the web server application. The MPEG-21 Conformant application is implemented as a plug-in application that will be downloaded to the user terminal when user accesses the webpage for the first time.

As the input experiment, we make a DID that encapsulates the resource(s) that can be consumed by the end user together with all the description data for each resource. Figure 5 shows the example of the license that is embedded in the DID to grant the consumption to a specific resource for the user with some conditions to be met prior to the consumption. To ensure that the license is obeyed, we write our DIM as shown in figure 6.

```
<?xml version="1.0" encoding="UTF-8"?>
<r:license xmlns:r="urn:mpeg:mpeg21:2003:01-REL-R-NS"
xmlns:sx="urn:mpeg:mpeg21:2003:01-REL-SX-NS"
xmlns:mx="urn:mpeg:mpeg21:2003:01-REL-MX-NS"
xmlns:dsig="http://www.w3.org/2000/09/xmldsig#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:mpeg:mpeg21:2003:01-REL-R-NS ..\schemas\rel-r.xsd
urn:mpeg:mpeg21:2003:01-REL-SX-NS ..\schemas\rel-sx.xsd
urn:mpeg:mpeg21:2003:01-REL-MX-NS ..\schemas\rel-mx.xsd">
    <r:grant>
        <r:keyHolder>
            <r:info>
                <dsig:KeyValue>
                    <dsig:RSAKeyValue>
                        <dsig:Modulus>KtdToQQyzA==</dsig:Modulus>
                        <dsig:Exponent>AQABAA==</dsig:Exponent>
                    </dsig:RSAKeyValue>
                </dsig:KeyValue>
            </r:info>
        </r:keyHolder>
        <mx:play/>
        <r:digitalResource licensePartID="VideoA">
            <r:nonSecureIndirect
URI="http://mccb.icu.ac.kr/KBSTestbed/News/NewsH.avi"/>
        </r:digitalResource>
        <r:allConditions>
            <r:validityInterval>
                <r:notBefore>2003-01-01T00:00:00</r:notBefore>
                <r:notAfter>2003-12-31T12:59:59</r:notAfter>
            </r:validityInterval>
        </r:allConditions>
    </r:grant>
</r:license>
```

Figure 5. Example of license used in the experiment.

```
function PlayFullContent( digitalItem )
{
    var resource = GetDIDLNode( "Component[1]/Resource[1]", digitalItem );
    var objectTypes = new Array("Node", "Node");
    if( resource != null ) {
      var license = RequestResourceAuthorization (resource, "mx:play");
      if (license) {
        var isPrincipalOk = AuthenticatePrincipal (license);
        var isConditionOk = ValidateRightsCondition (license, objectTypes,
                                    null);
        if (isPrincipalOk && isConditionOk) {
          PlayResource(resource, null, true);
        } else {
          Alert("Condition violation!!!");
        }
      } else {
        Alert("No license is found to play this content!!!");
      }
    }
}
```

Figure 6. Definition of PlayContent.

As the end user terminal receives the DI from the server, the MPEG-21 Conformant application will be launched. It will take action according to the code written in the DIMs. As shown in table 2, the DIM applies some code for license checking prior to display the content. Figure 7 shows the captured images from the application GUIs when it is running.

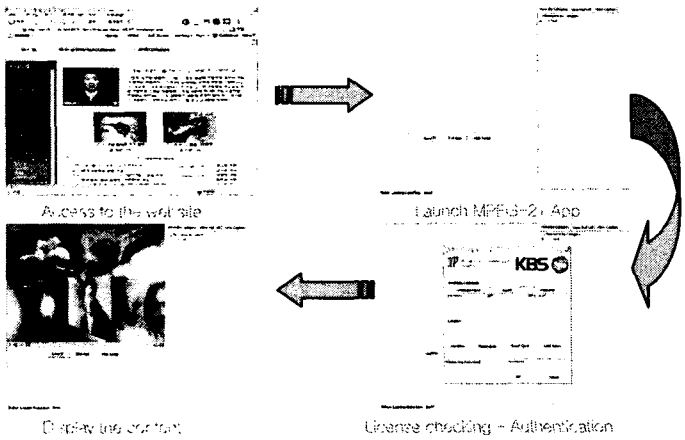

Figure 7. Application GUI.

4. Conclusion and Future Work

Combination of many parts of MPEG-21 framework can be utilized for many applications. In our test bed, we have shown that by using Digital Item Declaration, Rights Expression Language, and Digital Item Processing we can develop an authorized multimedia delivery and consumption application.

There are still many parts that we see potential to be included in our future test bed such as the MPEG-21 Intellectual Property Management and Protection (IPMP) to support secure and interoperable multimedia processing, MPEG-21 Digital Item Adaptation (DIA) to support adaptation of resource(s) so it can be played in various devices and environment, etc.

References:

[1] MPEG MDS Group, MPEG-21 Vision, Technology, and Strategy, ISO/IEC TC JTC1/SC29/WG11/N6269, Dec 2003, Waikoloa, HI.

[2] MPEG MDS Group, MPEG-21 Digital Item Declaration 2nd Edition Working Draft (WD) v.2, ISO/IEC JTC1/SC29/WG11/N6164, Dec 2003, Waikoloa, HI.

[3] MPEG MDS Group, MPEG-21 Rights Expression Language Final Draft for International Standard, ISO/IEC JTC1/SC29/WG11/N5839, Jul 2003, Trondheim, Norway.

[4] MPEG MDS Group, MPEG-21 Digital Item Processing Committee Draft, ISO/IEC JTC1/SC29/WG11/N6173, Dec. 2003, Waikoloa, HI.