INTERNATIONAL STANDARD

ISO/IEC 21000-5

First edition
2004-04-01

AMENDMENT 1
2007-02-01

Information technology — Multimedia framework (MPEG-21) —

Part 5:
Rights Expression Language

AMENDMENT 1: MAM (Mobile And optical Media) profile

Technologies de l'information — Cadre multimédia (MPEG-21) —

Partie 5: Langage d'expression des droits

AMENDEMENT 1: Profil MAM ("Mobile And optical Media")

Reference number
ISO/IEC 21000-5:2004/Amd.1:2007(E)

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

Amendment 1 to ISO/IEC 21000-5:2004 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

Information technology — Multimedia framework (MPEG-21) —

Part 5:
Rights Expression Language

AMENDMENT 1: MAM (Mobile And optical Media) profile

Insert two new clauses 10 and 11 as follows:

10 Additional Extensions

10.1 General

This clause lists extensions of the REL.

The REL (Rights Expression Language) defined in the previous clauses and Annexes A through J is a comprehensive, flexible and domain-agnostic rights language that can be used for a wide spectrum of applications. However, in order to meet specific needs on the REL from, for example, an ecosystem (e.g., electronic books or mobile devices), an application (e.g., media streaming), a function (e.g., certificating and attesting assertions), or a region (e.g., North America, Europe or Asia), the entire REL may need to be restricted to some profiles for the purposes of optimization and interoperability in developing and deploying systems, applications, components and devices. Here, a profile here represents a subset of the REL types, elements and attributes (with any specified syntactic, semantic and processing restrictions), possibly together with some further extensions to the REL types and some newly introduced elements and attributes, all subject to maintaining compliance to the specification of the REL.

10.2 Multimedia Extension One

10.2.1 General

This subclause specifies an additional extension, called the “multimedia extension one”, of the REL specified in the previous clauses.

10.2.2 Normative Namespace

The XML namespace for the extension types and new elements and attributes introduced in the REL multimedia extension one shall be `urn:mpeg:mpeg21:2005:01-REL-M1X-NS`, and this namespace is normative. Here, the `01` represents a serial number that is expected to change as this extension evolves.

10.2.3 Namespace Prefixes

For convenience, this extension uses shorthand namespace prefixes when referring to XML elements and types. The actual prefix used is not important as long as the namespace URI is correct. The prefixes used in this extension are given in Table AMD1.1.

Table AMD1.1 — Prefixes of XML Schemas

Prefix	Name	Namespace
`r`	REL Core	`urn:mpeg:mpeg21:2003:01-REL-R-NS`
`sx`	REL Standard Extension	`urn:mpeg:mpeg21:2003:01-REL-SX-NS`
`mx`	REL Multimedia Extension	`urn:mpeg:mpeg21:2003:01-REL-MX-NS`
`dsig`	XML Digital Signature	`http://www.w3.org/2000/09/xmldsig#`
`xenc`	XML Encryption	`http://www.w3.org/2001/04/xmlenc#`
`m1x`	REL Multimedia Extension One	`urn:mpeg:mpeg21:2005:01-REL-M1X-NS`

10.2.4 Defintion of Multimedia Extension One

This subclause defines the multimedia extension one to the REL. The syntax and the semantics of the types and elements in the multimedia extension one are presented here. The XML schema for the extension elements and types is listed in Annex L.

10.2.4.1 Principal Extension Elements

10.2.4.1.1 IdentityHolder

10.2.4.1.1.1 Informative Description

The `m1x:identityHolder` element is an extension element of the type `r:Principal` defined in the REL Core. It identifies the principal who is the holder of the specified identity possibly in some identification system. The optional `m1x:idSystem` child element of the `xsd:anyURI` type is used to indicate the identification system. The `m1x:idValue` child element, which can be either character content or element content from any namespace, is used to contain the identifier value of the identity.

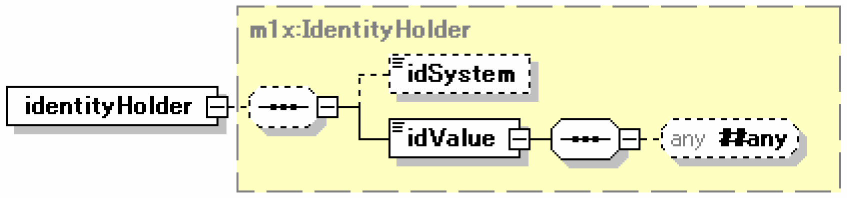

Figure AMD1.1 — `m1x:identityHolder` Principal

The following example specifies a principal as the holder of an International Mobile Subscriber Identifier.

```
<r:grant>
    <m1x:identityHolder>
        <idSystem>urn:acme:imsi</idSystem>
        <idValue>IMSI:2232111123</idValue>
    </m1x:identityHolder>
    <mx:play/>
    <r:digitalResource>
        <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
    </r:digitalResource>
</r:grant>
```

In the above example, the principal identified by `m1x:identityHolder` is granted the right to play the resource specified in `r:digitalResource`.

The types of the `m1x:identityHolder` element and the `m1x:idValue` child element are defined as extensions of the `r:LicensePart` type. Consequently, both `m1x:identityHolder` and `m1x:idValue` can be specified by variables. The following example is like the example above, except the principal is a holder of any, instead of a particular, International Mobile Subscriber Identifier.

```
<r:grant>
    <r:forAll varName="imsiIdentifier"/>
    <m1x:identityHolder>
        <idSystem>urn:acme:imsi</idSystem>
        <idValue varRef="imsiIdentifier"/>
    </m1x:identityHolder>
    <mx:play/>
    <r:digitalResource>
        <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
    </r:digitalResource>
</r:grant>
```

10.2.4.1.1.2 Normative Specification

Let *p* be a `m1x:IdentityHolder`. If *p*/`m1x:idValue` has a child element, then *p*/`m1x:idValue` identifies that system entity that possesses the identifier indicated by that child element. Otherwise, *p*/`m1x:idValue` identifies that system entity that possesses the identifier indicated by the value of *p*/`m1x:idValue`. In either case, the identifier belongs to the identification system indicated by the URI value of *p*/`m1x:idSystem` when the *p*/`m1x:idSystem` child element is present.

10.2.4.2 Right Extension Elements

10.2.4.2.1 GovernedCopy

10.2.4.2.1.1 Informative Description

This element represents the right to copy the resource and at the same time to result in certain rights being associated to the copied resource. The optional attribute `@m1x:governanceRule` of type `QName` indicates the name of a governance rule that determines how exactly the copy should be made and what rights should be associated and by whom for the copied resource. When the attribute is not specified, this right allows to make a bit-wise identical copy of the resource and to result in an identical copy of the `r:license` that this right is specified being made to the copied resource. This right can be used for copying the resource that is either locally available or received from a remote location (e.g., via streaming or broadcasting) during the time this right is being exercised.

Figure AMD1.2 — `m1x:governedCopy` Right

A sample code fragment is provided below for illustration:

```
<r:license>
    <r:grant>
        <mx:play/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
        </r:digitalResource>
    </r:grant>
    <r:grant>
        <m1x:governedCopy/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
        </r:digitalResource>
    </r:grant>
</r:license>
```

In the above example, any principal is granted the right to play a movie clip, and the right to copy the clip together with the same license.

The following is another example license.

```
<r:license >
    <r:grant>
        <mx:play/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
        </r:digitalResource>
    </r:grant>
    <r:grant>
        <m1x:governedCopy governanceRule="acme:CopyOnce"/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
        </r:digitalResource>
    </r:grant>
    <r:issuer>
```

```
        <r:keyHolder>
            <r:info>
                <dsig:KeyName>Rights Issuer Public Key Name</dsig:KeyName>
            </r:info>
        </r:keyHolder>
    </r:issuer>
</r:license >
```

Suppose the governance rule named “`acme:CopyOnce`” only allows exercising this right once to make a bit-wise identical copy of the resource and associating the other rights in the same license to the copied resource by issuing another license by the same issuer. In this case, exercising the right `m1x:governedCopy` in the license will result in a bit-wise identical copy of the resource, and the following license:

```
<r:license >
    <r:grant>
        <mx:play/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
        </r:digitalResource>
    </r:grant>
    <r:issuer>
        <r:keyHolder>
            <r:info>
                <dsig:KeyName>Rights Issuer Public Key Name</dsig:KeyName>
            </r:info>
        </r:keyHolder>
    </r:issuer>
</r:license >
```

10.2.4.2.1.2 Normative Specification

Let *r* be a `m1x:GovernedCopy`. Then, if *r*/`@m1x:governanceRule` is present, *r* identifies the act of making a copy and associating right expressions with that copy in compliance with the compliance rules identified by *r*/`@m1x:governanceRule`. Otherwise, if *r*/`@m1x:governanceRule` is absent, *r* identifies the act of making a bit-wise identical copy and associating a right expression to that copy that is Equal to the License in the authorizer in one of the authorization proofs for the authorization request for that copy.

If *r* is used as the Right Member of an authorization request, then the Resource Member of that authorization request shall be present and shall identify the resource being copied.

10.2.4.2.2 GovernedMove

10.2.4.2.2.1 Informative Description

This element represents the right to move the resource and at the same time to result in certain rights being associated to the moved resource. The moved resource and its associated rights should not exist or be valid on the original repository after exercising this right. The optional attribute `@m1x:governanceRule` of type `QName` indicates the name of a governance rule that determines how exactly the resource should be moved and what rights should be associated and by whom for the moved resource. When the attribute is not specified, this right allows to the moved resource to be a bit-wise identical copy of the resource and to result in an identical copy of the `r:license` that this right is specified being made to the moved resource. This right can be used for moving the resource that is either locally available or received from a remote location during the time this right is being exercised.

Figure AMD1.3 — `m1x:governedMove` Right

A sample code fragment is provided below for illustration.

```
<r:license>
    <r:grant>
        <mx:play/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
        </r:digitalResource>
    </r:grant>
    <r:grant>
        <m1x:governedMove/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
        </r:digitalResource>
    </r:grant>
</r:license>
```

In the above example, any principal is granted the right to play a movie clip, and the right to move the clip together with the same license.

Following is another example license.

```
<r:license >
    <r:grant>
        <mx:play/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
        </r:digitalResource>
    </r:grant>
    <r:grant>
        <m1x:governedMove governanceRule="acme:MoveOnce"/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
        </r:digitalResource>
    </r:grant>
    <r:issuer>
        <r:keyHolder>
            <r:info>
                <dsig:KeyName>Rights Issuer Public Key Name</dsig:KeyName>
            </r:info>
        </r:keyHolder>
    </r:issuer>
</r:license >
```

Suppose the governance rule named "`acme:MoveOnce`" only allows exercising this right once to move the resource in the bit-wise identical manner and associating the other rights in the same license to the moved resource by issuing another license by the same issuer. In this case, exercising the right `m1x:governedMove` in the license will result in moving the resource in the bit-wise identical manner and the following license:

```
<r:license >
    <r:grant>
        <mx:play/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
        </r:digitalResource>
    </r:grant>
    <r:issuer>
        <r:keyHolder>
            <r:info>
                <dsig:KeyName>Rights Issuer Public Key Name</dsig:KeyName>
            </r:info>
        </r:keyHolder>
    </r:issuer>
</r:license >
```

10.2.4.2.2.2 Normative Specification

Let *r* be a `m1x:GovernedMove`. Then, if *r*/`@m1x:governanceRule` is present, *r* identifies the act of performing the act `mx:move` to the Resource and associating right expressions with that moved Resource in compliance with the compliance rules identified by *r*/`@m1x:governanceRule`. Otherwise, if *r*/`@m1x:governanceRule` is absent, *r* identifies the act of performing the act `mx:move` to the Resource in the bit-wise identical sense and associating a right expression to that moved Resource that is Equal to the License in the authorizer in one of the authorization proofs for the authorization request for that moved Resource.

If *r* is used as the Right Member of an authorization request, then the Resource Member of that authorization request shall be present and shall identify the resource being moved.

10.2.4.2.3 Enlist

10.2.4.2.3.1 Informative Description

This element represents the right to link or enlist (the reference to) the related resource into a new playback control sequences description (i.e. play-list) for the optical disc. When it presents in an `r:grant` element, this element allows adding the value of the `resourceLocator` in the `m1x:protectedResource` element to the play-list.

Figure AMD1.4 — `m1x:enlist` Right

The following grant permits any user to link the associated (video object) resource “`file:///HDDVDdisc/ADV_OBJ/Vo3`” into a new play list being edited currently. In this example, the value of the `m1x:resourceLocator`, “`file:///HDDVDdisc/TM_OBJ/TM3`”, will be added to the play list when the user selects the object “`Vo3`”.

```
<r:grant>
   <m1x:enlist/>
   <m1x:protectedResource>
      <r:nonSecureIndirect URI="file:///HDDVDdisc/ADV_OBJ/Vo3"/>
      <resourceLocator>file:///HDDVDdisc/TM_OBJ/TM3</resourceLocator>
   </m1x:protectedResource>
</r:grant>
```

10.2.4.2.3.2 Normative Specification

Let *r* be a `m1x:enlist`. Then *r* identifies the act of adding the Resource into a Resource which is a list of Resources.

If *r* is used as the Right Member of an authorization request, then the Resource Member of that authorization request shall be present and shall identify the resource being added to the list Resource.

10.2.4.2.4 Delist

10.2.4.2.4.1 Informative Description

This element represents the right to unlink or delist (the reference to) the related resource from a related playback control sequences description (i.e. play-list) for the optical disc when the play-list is newly created from an existing one. When it presents in an `r:grant` element, this element allows to remove the value of the `resourceLocator` in the `m1x:protectedResource` element from a (new) play-list created from the associated resource described as a member of `m1x:isPartOf`.

Figure AMD1.5 — `m1x:delist` Right

The following grant permits any user to delist the associated (video object) resource “`file:///HDDVDdisc/ADV_OBJ/Vo3`” from any play-list created from the associated resource “`file:///cache/playlist003`". In this example, the value of the `m1x:resourceLocator`, “`file:///HDDVDdisc/TM_OBJ/TM3`”, will be removed from any newly created play-list that is created from the existing “`playlist003`” when the user selects to remove the “`Vo3`”.

```
<r:grant>
    <m1x:delist/>
     <m1x:protectedResource>
        <r:nonSecureIndirect URI="file:///HDDVDdisc/ADV_OBJ/Vo3"/>
        <resourceLocator>file:///HDDVDdisc/TM_OBJ/TM3</resourceLocator>
    </m1x:protectedResource>
    <m1x:derivationConstraint>
        <m1x:isPartOf>
            <m1x:protectedResource>
                <r:nonSecureIndirect URI= "file:///cache/playlist003"/>
            </m1x:protectedResource>
         </m1x:isPartOf>
    </m1x:derivationConstraint>
</r:grant>
```

10.2.4.2.4.2 Normative Specification

Let *r* be a `m1x:delist`. Then *r* identifies the act of removing the Resource from a Resource which is a list of Resources and contains the Resource to be removed. If a condition `m1x:derviationContraint` is present, is satisified and contains a child element `m1x:isPartOf`, then the Resource should be removed from Resource identified by the `m1x:isPartOf`, as the result of performing the act identified by *r*.

If *r* is used as the Right Member of an authorization request, then the Resource Member of that authorization request shall be present and shall identify the resource to be removed.

10.2.4.3 Resource Extension Elements

10.2.4.3.1 ProtectedResource

10.2.4.3.1.1 Informative Description

This resource represents a piece of content that is protected with some form of (symmetric key and/or public key) encryption. The element contains the following child elements: `r:digitalResource`, `xenc:EncryptedData`, and `xenc:EncryptedKey`.

The `r:digitalResource` element specifies the resource under consideration. The `nonSecureIndirect` element is used to identify the resource. If it is necessary to bind the identifier to the actual content of the resource, then the `secureIndirect` element provides elements to include a secure hash (`ds:DigestValue`) of the content.

The `xenc:EncrytedData` is for carrying information about encryption of the resource identified by `r:digitalResource`. The `xenc:EncryptedKey` contains information about encryption of the key (or keys) that is used to encrypt the resource. It should be clear that, regardless of whether the `xenc:EncryptedData` or `xenc:EncryptedKey` child elements appear, the `r:digitalResource` child element always represents the clear-text resource and not the encrypted resource.

The `m1x:resourceLocator` specifies the location of the associated resource. When the associated right is either `m1x:enlist` or `m1x:delist`, the location specified by this element should be added to or removed from the related containing resource. This element is different from the `r:digitalResource` element in that the latter provides the information for identifying the resource.

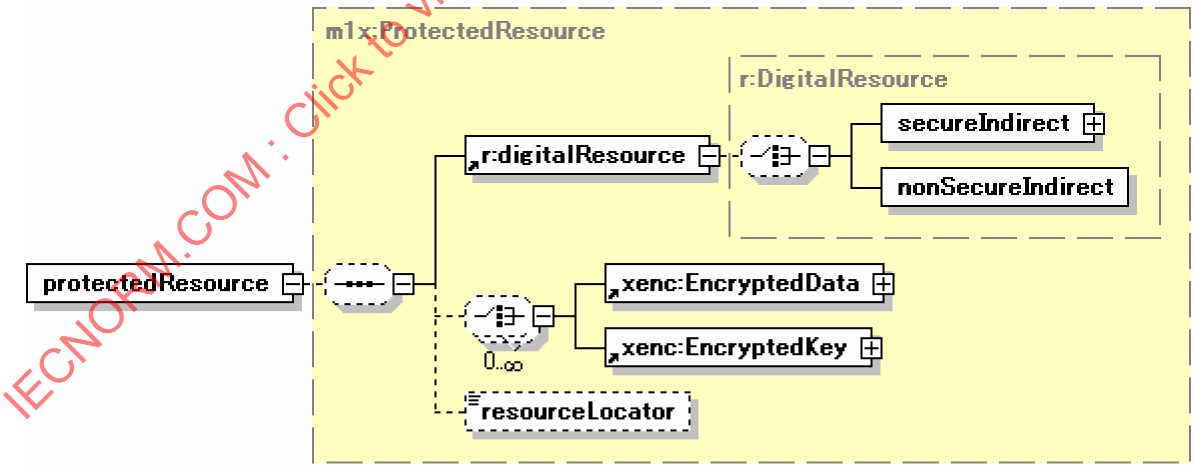

Figure AMD1.6 — `m1x:protectedResource` Resource

Here is an example showing that an MPEG movie resource is encrypted using the AES-128 encryption algorithm with a key whose key name is “Content Encryption Key Name”, and the AES key is further encrypted using the RSA encryption algorithm with a key whose key name is “User Key Name” and the encrypted key value is “AQABAA==”. This way the user can use a key with key name “User Key Name” to decrypt the resource encryption key from the value “AQABAA==” and use the decrypted key to further decrypt the encrypted movie resource.

```
<r:grant>
    <m1x:identityHolder>
        <idSystem>urn:acme:imsi</idSystem>
        <idValue>IMSI:2232111123</idValue>
    </m1x:identityHolder>
    <mx:play/>
    <m1x:protectedResource>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:encrypted-movie:matrixrevolutions.mpeg"/>
        </r:digitalResource>
        <xenc:EncryptedData Id="ED">
            <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#kw-aes128"/>
            <dsig:KeyInfo>
                <dsig:RetrievalMethod URI="#EK" Type="http://www.w3.org/2001/04/xmlenc#EncryptedKey"/>
                <dsig:KeyName>Content Encryption Key Name</dsig:KeyName>
            </dsig:KeyInfo>
        </xenc:EncryptedData>
        <xenc:EncryptedKey Id="EK">
            <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-1_5"/>
            <dsig:KeyInfo>
                <dsig:KeyName>User Key Name</dsig:KeyName>
            </dsig:KeyInfo>
            <xenc:CipherData>
                <xenc:CipherValue>AQABAA==</xenc:CipherValue>
            </xenc:CipherData>
            <xenc:ReferenceList>
                <xenc:DataReference URI="#ED"/>
            </xenc:ReferenceList>
            <xenc:CarriedKeyName>Content Encryption Key Name</xenc:CarriedKeyName>
        </xenc:EncryptedKey>
    </m1x:protectedResource>
</r:grant>
```

10.2.4.3.1.2 Normative Specification

Let t be an `r:ProtectedResource`. Then t identifies a Digital Resource whose plaintext sequence of digital bits b is the sequence of digital bits identified by t/`r:digitalResource`. Moreover,

— If t/`xenc:EncryptedData` is present, then it should represent the encryption of b per XMLENC.

— If t/`xenc:EncryptedKey` is present, then it should represent the encryption of the cryptographic keys used for encrypting b per XMLENC.

— If t`/resourceLocator` is present, the URI the value of t`/resourceLocator` provides the location of the digital bits b. When the associated right is either `m1x:enlist` or `m1x:delist`, the location specified by this element should be added to or removed from the related containing resource.

10.2.4.4 Condition Extension Elements

10.2.4.4.1 DerivationConstraint

10.2.4.4.1.1 Informative Description

This condition element is used for derivation, including adapting, diminishing, enhancing, playing, enlisting and delisting of the associated resource. It imposes possible restrictions on the following:

- which resource the associated resource has to be part of in order for the derivation to take place;
- which additional resources must or must not become part of the resource into which the associated resource is derived, and what possible temporal relationships of these additional resources must or must not have with the associated resource; and
- which resources can be used to replace the associated resource in the derived resource.

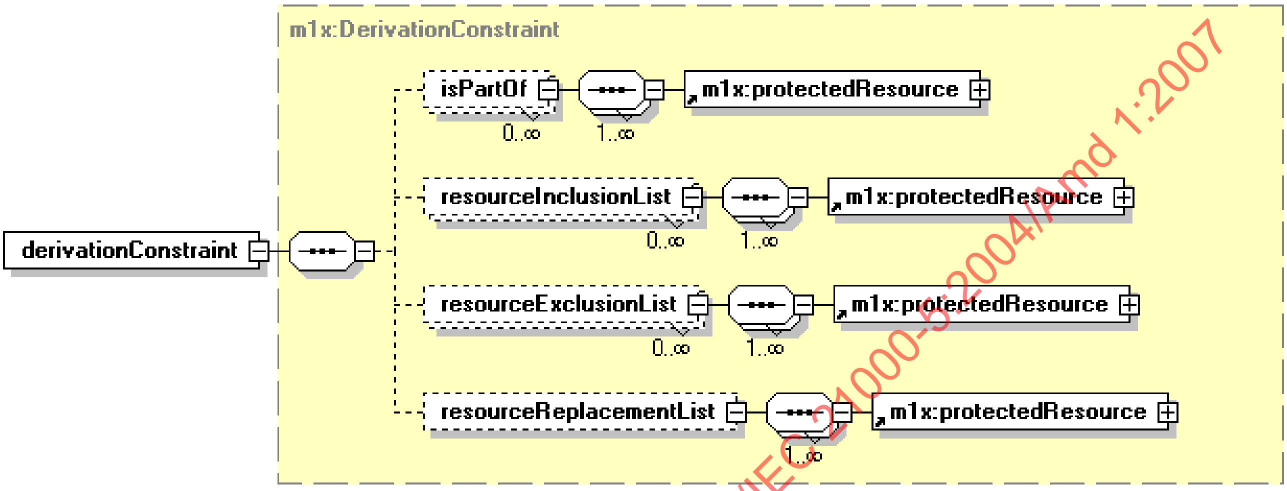

Figure AMD1.7 — `m1x:derivationConstraint` Condition

The element `m1x:isPartOf` indicates a list of protected resources, one or more of which the associated resource must be part of.

The element `m1x:resouceInclusionList` indicates a list of resources, each of which must become, possibly with respect to some temporal relations, part of the resource into which the associated resource is derived. The element `m1x:resouceExclusionList` indicates a list of resources, each of which must not become, possibly with respect to some temporal relations, part of the resource into which the associated resource is enhanced. The temporal relations are specified by values of an optional attribute `temporalRelation` of the two child elements. The attribute can take one of the following values:

- `any`: when rendered temporally, each of the list of resources is related to the associated resource in an unrestricted manner;
- `before`: when rendered temporally, each resource in the list is before the associated resource;
- `after`: when rendered temporally, each resource in the list is after the associated resource; and
- `simultaneous`: when rendered temporally, each resource in the list is simultaneous with the associated resource.

When the attribute is omitted, the default relation is identical to the one specified by the value “`any`” of the attribute.

The element `m1x:resourceReplacementList` indicates a list of resources, one of which must be used to replace the associated resource in the resource into which the associated resource is derived.

The condition specified by the `m1x:derivationConstraint` is satisfied, only if all of the following hold:

— If m1x:isPartOf is present, the associated resource is part of one or more resources listed in "m1x:isPartOf";

— if m1x:resourceInclusionList is present, the derived resource contains all of the resources listed in m1x:resourceInclusionList element, and all the resources being contained in the derived resource satisfy the respective temporal relation with the associated resource;

— if m1x:resourceExclusionList is present, the derived resource does not contain any of the resources listed in m1x:resourceExclusionList with respect to its temporal relation with the associated resource; and

— if m1x:resourceReplacementList is present, the derived resource contains the associated resource as the replacement of one of the resources which is contained in the derived resource and also listed in m1x:resourceReplacementList.

The following grant permits the associated (video object) resource urn:myPlaylist:vo:0 to be enlisted, but the enlisting must include the resource urn:myPlaylist:vo:1, must include the resource urn:myPlaylist:vo:2 before the associated resource, must not include the resource urn:myPlaylist:vo:3, and must not include the resource urn:myPlaylist:vo:4 simultaneously with the associated resource (which however does not forbid possibly before or after the associated resource). Because all of the individual resources have their locators specified by the m1x:resourceLocator elements, the individual locators are enlisted into the playlist.

```
<r:grant>
   <m1x:enlist/>
   <m1x:protectedResource>
         <r:nonSecureIndirect URI="urn:myPlaylist:vo:0"/>
         <resourceLocator>file:///HDDVDdisc/TM_OBJ/TM0</resourceLocator>
   </ m1x:protectedResource>
   <m1x:derivationConstraint>
      <m1x:resourceInclusionList>
            < m1x:protectedResource>
                  <r:nonSecureIndirect URI="urn:myPlaylist:vo:1"/>
                  <resourceLocator>file:///HDDVDdisc/TM_OBJ/TM1</resourceLocator>
             </ m1x:protectedResource>
      </m1x:resourceInclusionList>
      <m1x:resourceInclusionList temporalRelation="before">
             <m1x:protectedResource>
                  <r:nonSecureIndirect URI="urn:myPlaylist:vo:2"/>
                  <resourceLocator>file:///HDDVDdisc/TM_OBJ/TM2</resourceLocator>
            </m1x:protectedResource>
      </m1x:resourceInclusionList>
      <m1x:resourceExclusionList>
            <m1x:protectedResource>
                  <r:nonSecureIndirect URI="urn:myPlaylist:vo:3"/>
                  <resourceLocator>file:///HDDVDdisc/TM_OBJ/TM3</resourceLocator>
            </m1x:protectedlResource>
      </m1x:resourceExclusionList>
      <m1x:resourceExclusionList temporalRelation="simultaneous">
            <m1x:protectedResource>
                  <r:nonSecureIndirect URI="urn:myPlaylist:vo:4"/>
                  <resourceLocator>file:///HDDVDdisc/TM_OBJ/TM4</resourceLocator>
            </m1x:protectedResource>
        </m1x:resourceExclusionList>
   </m1x:derivationConstraint>
</r:grant>
```

According to the grant above, the following two Resources as the results of enlisting `vo:0` with others satisfy the `m1x:derivationConstraint`.

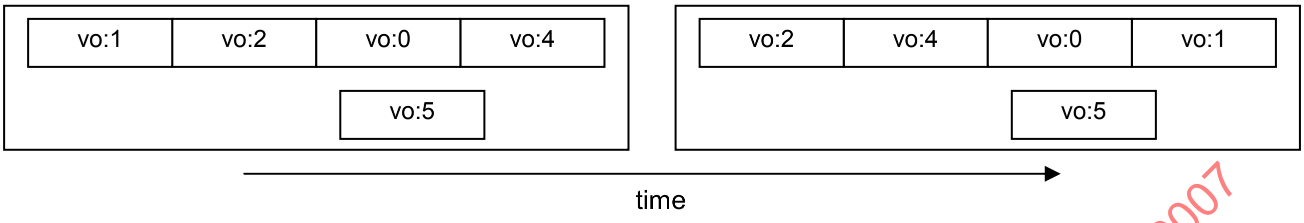

Figure AMD1.8 — Two correctly derived Resources

However, the following two Resources that enlist `vo:0` do not satisfy the `m1x:derivationConstraint`, because `vo:3` is included and `vo:4` is simultaneous with `vo:0` in the first Resource, and `vo:1` is not included and `vo:2` is after `vo:0` in the second Resource.

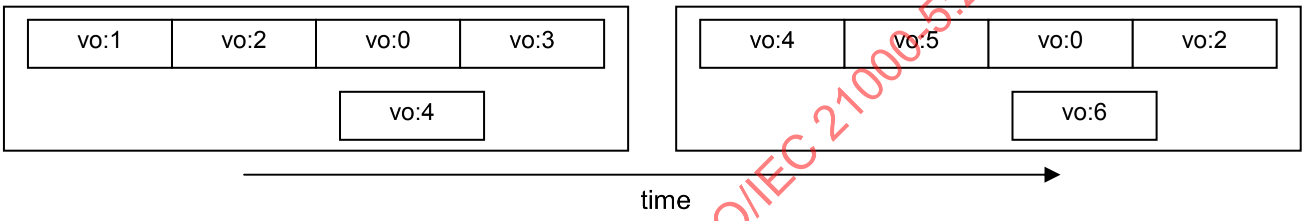

Figure AMD1.9 — Two incorrectly derived Resources

Another example is shown in Figure AMD1.10.

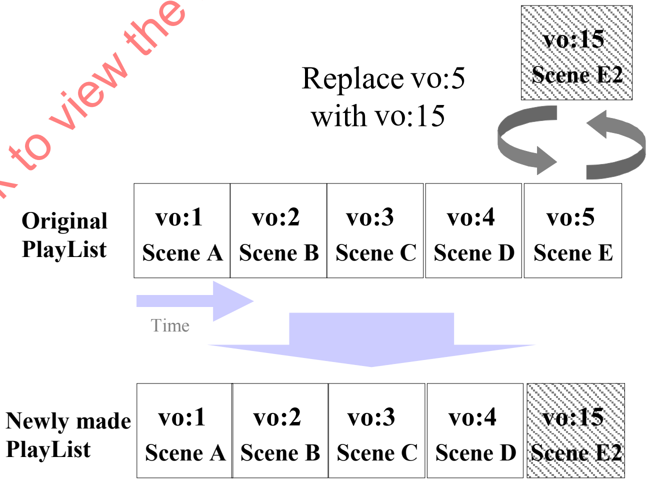

Figure AMD1.10 — Derivation with `m1x:resourceReplacementList`

According to the figure, the following grants permit the associated (video object) resource `urn:Playlist:vo:5` to be delisted to create a new containing resource, but the delisting must contain the replacement of `urn:voParts:vo:15` with `urn:Playlist:vo:5`. The `urn:Playlist:vo:5` itself is delisted from the new containing resource and any of the temporal relations and spatial relations of it with the rest of the original containing resource is inherited by `urn:voParts:vo:15`. Because all of the individual resources have no locators specified by the `m1x:resourceLocator` elements, the URIs of the individual resources are operated directly.

```
<r:grant>
    <m1x:delist/>
    <m1x:protectedResource>
        <r:nonSecureIndirect URI="urn:Playlist:vo:5"/>
        <resourceLocator>file:///HDDVDdisc/TM_OBJ/TM5</resourceLocator>
    </r:digitalResource>
    <m1x:derivationConstraint>
        <m1x:isPartOf>
            <m1x:protectedResource>
                        <r:nonSecureIndirect URI=" urn:titles:Playlist"/>
            </m1x:protectedResource>
         </m1x:isPartOf>
        <m1x:resoureReplacementList>
            <m1x:protectedResource>
                  <r:nonSecureIndirect URI="urn:voParts:vo:15"/>
                  <resourceLocator>file:///HDDVDdisc/TM_OBJ/TM15</resourceLocator>
            </m1x:protectedResource>
        </m1x:resoureReplacementList >
    </m1x:derivationConstraint>
</r:grant>
```

```
<r:grant>
    <m1x:enlist/>
    <m1x:protectedResource>
        <r:nonSecureIndirect URI="urn:voParts:vo:15"/>
       <resourceLocator>file:///HDDVDdisc/TM_OBJ/TM15</resourceLocator>
    </ m1x:protectedResource>
</r:grant>
```

10.2.4.4.1.2 Normative Specification

Let c be a `m1x:DerivationConstraint`. Let $(p, r, t, v, \Sigma, L, R)$ be an authorization request. Let (g, h, e) be an authorization story. Then c is Satisfied with respect to $(p, r, t, v, \Sigma, L, R)$ and (g, h, e) if and only if all of the following are true:

— if "c/`m1x:isPartOf`" is present, the associated Resource is part of at least one of the resources listed in "`m1x:isPartOf`";

— if c/`m1x:resourceInclusionList` is present then one of the following is the case, depending on the value of the attribute c/`m1x:resourceInclusionList/@temporalRelation`:

- if c/`m1x:resourceInclusionList/@temporalRelation` is omitted or present with its value equal to "`any`", then every protected Resource element in c/`m1x:resourceInclusionList` is a member of Σ.`m1x:eRL`(t);
- if c/`m1x:resourceInclusionList/@temporalRelation` is present with its value equal to "`before`", then every protected Resource element in c/`m1x:resourceInclusionList` is a member of Σ.`m1x:eRL`(t), and is temporally before t when the derived Resource is rendered;
- if c/`m1x:resourceInclusionList/@temporalRelation` is present with its value equal to "`after`", then every protected Resource element in c/`m1x:resourceInclusionList` is a member of Σ.`m1x:eRL`(t), and is temporally after t when the derived Resource is rendered; and

- if *c*/`m1x:resourceInclusionList/@temporalRelation` is present with its value equal to “`simultaneous`”, then every digital Resource element in *c*/`m1x:resourceInclusionList` is a member of *Σ*.`m1x:eRL`(*t*), and is temporally neither before nor after *t* when the derived Resource is rendered;

NOTE *c*/`m1x:resourceInclusionList` may be ignored in case that the associated resource or its replacement does not become part of the derived resource, for example as the result of exercising the `mx:diminish` or `m1x:delist` right.

— If *c*/`m1x:resourceExclusionList` is present, then one of the following is the case, depending on the value of the attribute *c*/`m1x:resourceInclusionList/@temporalRelation`:

- if *c*/`m1x:resourceExclusionList/@temporalRelation` is omitted or present with its value equal to “`any`”, then each protected Resource element in *c*/`m1x:resourceExclusionList` is not a member of *Σ*.`m1x:eRL`(*t*);

- if *c*/`m1x:resourceExclusionList/@temporalRelation` is present with its value equal to “`before`”, then each protected Resource element in *c*/`m1x:resourceExclusionList` is not a member of *Σ*.`m1x:eRL`(*t*) that is temporally before *t* when the derived Resource is rendered;

- if *c*/`m1x:resourceExclusionList/@temporalRelation` is present with its value equal to “`after`”, then each protected Resource element in *c*/`m1x:resourceExclusionList` is not a member of *Σ*.`m1x:eRL` (*t*) that is temporally after *t* when the derived Resource is rendered; and

- if *c*/`m1x:resourceExclusionList/@temporalRelation` is present with its value equal to “`simultaneous`”, then each protected Resource element in *c*/`m1x:resourceExclusionList` is not a member of *Σ*.`m1x:eRL`(*t*) that is temporally neither before nor after *t* when the derived Resource is rendered;

NOTE *c*/`m1x:resourceExclusionList` may be ignored in case that the associated resource or its replacement does not become part of the derived resource, for example as the result of exercising the `mx:diminish` or `m1x:delist` right.

— if “*c*/`m1x:resourceReplacementList`” is present, then one and only one protected Resource element in *c*/`m1x:resourceReplacementList` is Equal to the Resource element specified by *Σ*.`m1x:rR`(*t, p*).

NOTE *c*/`m1x:resourceReplacementList` may be ignored in case that the associated resource itself does become part of the derived resource, for example as the result of exercising the `mx:enhance` or `m1x:enlist` right.

10.2.4.4.2 DrmSystem

10.2.4.4.2.1 Informative Description

This condition element is used in conjunction with the `m1x:GovernedCopy` and `m1x:GovernedMove` rights. It places a condition on the right to move or copy the specified digital resource in terms of which class of external DRM system the resource may be exported to. A user may only export to the class of systems specified in this element.

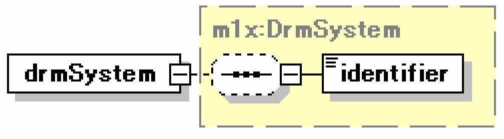

Figure AMD1.11 — `m1x:drmSystem` Condition

This element contains a sub-element called `m1x:identifier`, which is a URI that uniquely identifies the class of DRM systems. An example is provided for illustration:

```
<r:grant>
    <m1x:identityHolder>
        <idSystem>urn:acme:imsi</idSystem>
        <idValue>IMSI:2232111123</idValue>
    </m1x:identityHolder>
    <m1x:governedCopy/>
    <r:digitalResource>
        <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
    </r:digitalResource>
    <m1x:drmSystem>
        <mpx:identifier>urn:ACME_STORAGEMEDIA_SYSTEM:Version10</mpx:identifier>
    </m1x:drmSystem>
</r:grant>
```

10.2.4.4.2.2 Normative Specification

Let c be a `m1x:DrmSystem`. Let $(p, r, t, v, \Sigma, L, R)$ be an authorization request. Let (g, h, e) be an authorization story. Then c is Satisfied with respect to $(p, r, t, v, \Sigma, L, R)$ and (g, h, e) if and only if the value c/`m1x:identifier` is Equal to Σ.`m1x:drmSystem`().

10.2.4.4.3 OutputRegulation

10.2.4.4.3.1 Informative Description

This condition element requires output signal to be regulated using any of the regulations specified by the list of `m1x:regulation` elements.

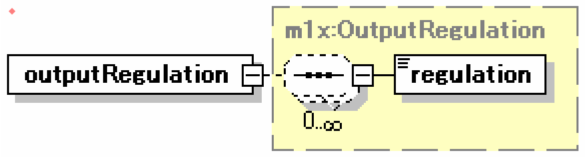

Figure AMD1.12 — `m1x:outputRegulation` Condition

The optional attribute `@m1x:typeOfSignal` of `m1x:regulation` indicates which type, `m1x:digital` or `m1x:analog`, of signal the regulation applies. When this attribute is not present, the regulation applies to any type. The optional attribute `@m1x:qualityOfSignal` of `m1x:regulation` indicates which quality, `m1x:HD` (for high-definition) or `m1x:SD` (for standard definition), of the signal the regulation applies. When this attribute is not present, the regulation applies to any quality of the signal.

This condition is satisfied only if at least one of the regulations specified by the list of `m1x:regulation`s is used to regulate the output signal with a matched type and matched quality. Here, the type of the signal matches with the type of the regulation if the associated `m1x:regulation`s has either no type specified or an identical type, and the quality of the signal matches with the quality of the regulation if the associated `m1x:regulation` has either no quality specified or an identical quality.

The following example shows that a movie trailer is allowed to play when the output signal is regulated by either allowing High Definition Analog Output in the form of Constrained Image (ICT:1) or having Analog Protection according to Type 1 of APS (APSTB:01).

```
<r:grant>
    <mx:play/>
    <r:digitalResource>
        <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
    </r:digitalResource>
    <m1x:outputRegulation>
        <m1x:regulation typeOfSignal="m1x:analog" qualityOfSignal="m1x:HD">ICT:1</m1x:regulation >
        <m1x:regulation typeOfSignal="m1x:analog">APSTB:01</m1x:regulation>
    </m1x:outputRegulation >
</r:grant>
```

10.2.4.4.3.2 Normative Specification

Let c be a `m1x:OutputRegulation`. Let $(p, r, t, v, \Sigma, L, R)$ be an authorization request. Let (g, h, e) be an authorization story. Then c is Satisfied with respect to $(p, r, t, v, \Sigma, L, R)$ and (g, h, e) if and only if, for every integer i from 1 to Σ.m1x:oRNum(), there exists a c/`m1x:regulation` child γ of c such that all of the following are true:

— γ/`@m1x:typeOfSignal` is absent or its value is Equal to Σ.m1x:oRTOS(i),

— γ/`@m1x:qualityOfSignal` is absent or its value is Equal to Σ.m1x:oRQOS(i), and

— Σ.m1x:oR(i, the value of γ) is true.

10.2.4.4.4 SeekPermission and ServiceLocation

10.2.4.4.4.1 Informative Description

This condition element requires that permission from a server be sought before the associated right may be exercised, and restricts a time period during which an obtained permission can be cached for future use without contacting the server.

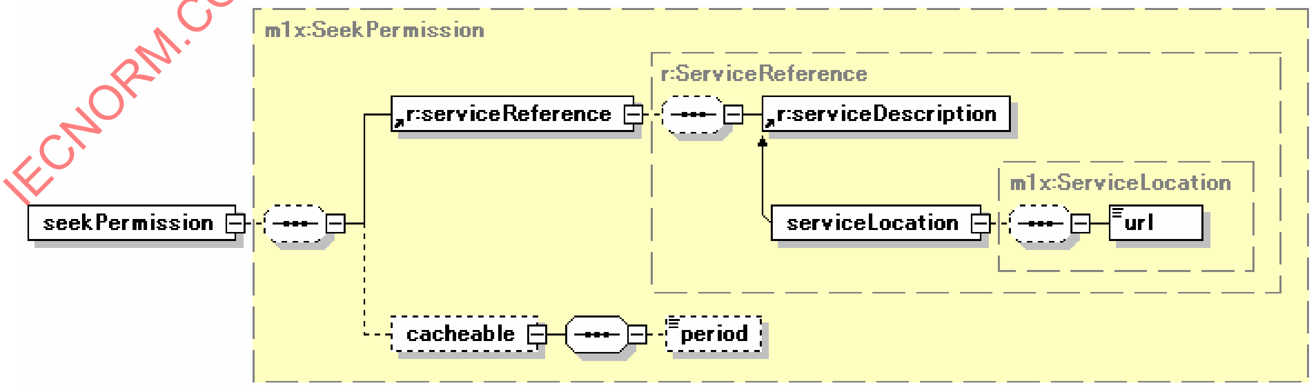

Figure AMD1.13 — `m1x:seekPermission` Condition

The `r:serviceReference` element, when used in the `m1x:seekPermission` element, describes a reference to a server from which the permission for exercising the associated right must be sought. The `m1x:serviceLocation` specifies a server by its location `m1x:url` indicating where the server is located.

The optional `m1x:cacheable` element is used to indicate that the permission obtained from the server may be cached. Its child element `m1x:period` indicates the amount of time that the permission may stay in the cache until it must be deleted.

The condition specified by the element is satisfied only when any of the following is true:

1. The element `m1x:cacheable` is present, and there is a permission in the cache that grants the associated right to be exercised.

2. The element `m1x:cacheable` is not present, the permission is obtained from the server that grants the associated right to be exercised.

In the following example, the right to play a video object can be exercised only if permission is obtained from the server at “`http://www.foo.org/paymentService`”.

```
<r:grant>
    <mx:play/>
    <r:digitalResource>
        <r:nonSecureIndirect URI="urn:myPlaylist:evobs:1"/>
    </r:digitalResource>
    <m1x:seekPermission>
        <r:serviceReference>
            <m1x:serviceLocation>
                <m1x:url>http://www.foo.org/paymentService</m1x:url>
            </m1x:serviceLocation>
        </r:serviceReference>
    </m1x:seekPermission>
</r:grant>
```

10.2.4.4.4.2 Normative Specification

Let c be a `m1x:SeekPermission`. Let (p, r, t, v, Σ, L, R) be an authorization request. Let (g, h, e) be an authorization story. Let m be c/`r:serviceReference`. Then c is Satisfied with respect to (p, r, t, v, Σ, L, R) and (g, h, e) if and only if, letting ρ be the ordered tuple containing the values of the reference-specific parameters determined by m, at least one of the following is true:

— Σ.m1x:sP(m/`r:serviceDescription`, ρ) is true or

— all of the following are true for σ equal to some subset of Σ:

  — c/`m1x:cacheable` is present,

  — Σ.m1x:sPC(m/`r:serviceDescription`, ρ, p, r, t, σ) exists, and

  — if c/`m1x:cacheable`/`m1x:period` is present, then Σ.m1x:sPC(m/`r:serviceDescription`, ρ, p, r, t, σ) is less than the value of c/`m1x:cacheable`/`m1x:period`.

Let d be a `m1x:ServiceLocation`. The endpoint of the service is given by the value of d/`m1x:url`.

10.2.4.4.5 StartCondition

10.2.4.4.5.1 Informative Description

This condition element requires the contained condition be checked at the start of an exercise of the associated right.

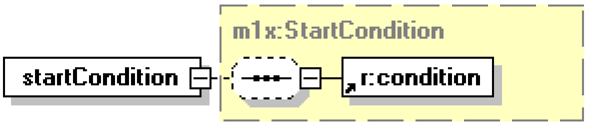

Figure AMD1.14 — `m1x:startCondition` Condition

The condition is satisfied only if the contained condition is satisfied at the starting time of exercising the associated right.

Using this condition to wrap another condition (such as a time condition) makes it possible to satisfy this condition without necessarily knowing how long the requested exercise will last in order to test the wrapped condition and without having to continually check the wrapped condition (which otherwise may be required) as the requested exercise continues to take place.

For example, the following expression specifies that the resource can be played as long as the playing starts within the year of 2005.

```
<r:grant>
    <mx:play/>
    <r:digitalResource>
        <r:nonSecureIndirect URI="urn:myPlaylist:evobs:1"/>
    </r:digitalResource>
    <m1x:startCondition licensePartId="startIn2005">
        <r:validityInterval>
            <r:notBefore>2005-01-01T00:00:00</r:notBefore>
            <r:notAfter>2005-12-31T23:59:59</r:notAfter>
        </r:validityInterval>
    </m1x:startCondition>
</r:grant>
```

10.2.4.4.5.2 Normative Specification

Let c be a `m1x:StartCondition`. Let (p, r, t, v, Σ, L, R) be an authorization request. Let (g, h, e) be an authorization story. Then c is Satisfied with respect to (p, r, t, v, Σ, L, R) and (g, h, e) if and only if c/`r:condition` is Satisfied with respect to (p, r, t, i, Σ, L, R) and (g, h, e) where i is the interval of zero length starting at the start of time interval v.

10.2.4.5 License Attribute Extensions

10.2.4.5.1 LicenseType

10.2.4.5.1.1 Informative Description

The attribute `@m1x:licenseType` of `r:License` specifies a list of QNames, each of which provides a further categorization of the License in addition to any profile it conforms to, which is useful in identifying what elements and attributes the License may contain.

10.2.4.5.1.2 Normative Specification

For an `r:License` *l*, each list member in the value of the *l*/`@m1x:licenseType` attribute, if the attribute is present, shall provide a Qualified Name representing one type to which the License belongs. If present, this attribute need not provide a complete list of the types to which the License belongs. Applications may ignore this attribute. If they do process it, they may ignore any members of the list.

10.2.4.6 Authorization Context Properties

Table AMD1.2 specifies the authorization context properties relating to the multimedia extention-1 and the statements they represent. If a property has the name given in the first column of Table AMD1.2 and the value given in the second column of Table AMD1.2, then the statement represented by that property is the statement given in the third column of Table AMD1.2.

Table AMD1.2 — Multimedia extension one authorization context properties

Property name	Property value	Statement represented
m1x:drmSystem()	U	*u* is an `xsd:anyURI`, and *u* identifies the class of the DRM system the destination repository (according to the semantics of the Right Member of the authorization request) for the requested performance..
m1x:eRL(t)	*D*	t is an `m1x:protectedResource` and *D* is a set of `m1x:protectedResource` elements used for deriving t during a derivation of t.
m1x:rR(t, p)	t'	t is an `m1x:protectedResource` element, *p* is an `m1x:protectedResource` in which *t* is included as a part, t' is another `m1x:protectedResource` element, and t' is an replacement of t during a derivation of *p*.
m1x:oR(i, q)	True	i is an integer, q is an `xsd:QName`, and q identifies one of the output regulations applied to the i^{th} output signal used in the requested performance.
m1x:oRNum	I	i is an integer and i is the total number of output signals used in the requested performance.
m1x:oRQOS(i)	Q	i is an integer, q is an `xsd:QName`, and q identifies the quality of the i^{th} output signal used in the requested performance.
m1x:oRTOS(i)	Q	i is an integer, q is an `xsd:QName`, and q identifies the type of the i^{th} output signal used in the requested performance.
m1x:sP(d, ρ)	True	d is an `r:ServiceDescription`, ρ is an ordered tuple, and the service described by d claims that this property may be used in an authorization context to establish permission for the requested performance.
m1x:sPC(d, ρ, p, r, t, σ)	δ	d is an `r:ServiceDescription`, ρ is an ordered tuple, p is an `r:Principal`, r is an `r:Right`, t is an `r:Resource`, σ is an authorization context, δ is a non-negative duration, and there is cache record indicating that the service described by d generated an affirmative response with respect to ρ, p, r, t, and σ at a time occurring before the start of the requested performance by a duration of δ.

10.2.4.7 Qualified Names

10.2.4.7.1 The REL Multimedia Extenstion-1 QName

The qualified name `RELProfile:MAM` is the attribute value of `r:license/@sx:profileCompliance`. It is used to indicate that it is the REL MAM profile that the license is compliant to.

10.2.4.7.2 Type-of-Signal QNames

10.2.4.7.2.1 Analog

The qualified name `m1x:analog` identifies the analog type of signal.

10.2.4.7.2.2 Digital

The qualified name `m1x:digital` identifies the digital type of signal.

10.2.4.7.3 Quality-of-Signal QNames

10.2.4.7.3.1 SD

The qualified name `m1x:SD` identifies the standard definition quality of signal.

10.2.4.7.3.2 HD

The qualified name `m1x:HD` identifies the high definition quality of signal.

11 REL Profiles

11.1 General

This clause defines profiles of the REL as specified in the previous clauses. An REL profile contains a subset, , of elements and types in the REL as defined in the previous clauses (including the those in the Additional Extensions clause), together with possibly further restrictions on their syntax and semantics.

11.2 MAM Profile

11.2.1 General

This subclause specifies a profile, called the “MAM (Mobile And optical Media) profile”, of the REL specified in the previous clauses.

11.2.2 MAM Profile Definition

The profile elements and their included child elements are listed in Table AMD1.3.

Table AMD1.3 — Profile Elements

Element/Child Element	Comments
`r:license`	An `r:license` is restricted to contain the following elements: `r:grant`, and `r:issuer`.
`r:grant`	Each `r:grant` represents a statement about what principal has what right over what resource under what condition.
`r:issuer`	This element indicates which principal issues the license. Restricted to at most one occurrence from unbounded occurrences.
`@sx:profileCompliance` `@m1x:licenseType`	The `@sx:profileCompliance` attribute indicates a profile that the license is compliant to. The value of `RELProfile:MAM` shall be used in a license to indicate compliance to this profile. The attribute `@m1x:licenseType` provides a further categorization of the license, which is useful in identifying what elements and attributes the license may contain.
`r:grant`	An `r:grant` is restricted to include the following child elements only: `r:forAll`, `r:principal`, `r:right`, `r:resource` and `r:condition`.

`r:anXmlExpression`
`r:forAll`
`r:propertyPossessor`
	These elements are used to identify a category of entities (resources, principals, etc.) that match an expression or possess a property. This is useful for specifying memberships, subscriptions, or domains.

`r:keyHolder` `m1x:identityHolder`	This profile only supports these `r:Principal`s.

	r:issue r:possessProperty mx:execute mx:play mx:print m1x:governedCopy m1x:governedMove m1x:enlist m1x:delist	This profile only supports these `r:Right`s.
	r:digitalResource sx:propertyUri m1x:protectedResource	This profile only supports these `r:Resource`s.
	r:allConditions r:validityInterval sx:exerciseLimit sx:territory sx:validityIntervalFloating sx:validityTimeMetered m1x:derivationConstraint m1x:drmSystem m1x:outputRegulation m1x:seekPermission m1x:startCondition	This profile only supports these `r:Condition`s.
r:propertyPossessor		This element is restricted to include the following child elements only: `r:propertyUri` and `r:trustedRootIssuers`.
	r:propertyUri	A property using an URI. Used as defined.
	r:trustedRootIssuers	Trusted root issuers. Used as defined.
r:keyHolder		This element is used as defined.
	r:info	This element of type `dsig:KeyInfoType` specifies a key for the `r:keyHolder`. Used as defined.
m1x:identityHolder		This element is used as defined.
r:digitalResource		This element is restricted to include the following child elements only: `r:nonSecureIndirect` and `r:secureIndirect`.
	r:nonSecureIndirect r:secureIndirect	`r:nonSecureIndirect` identifies a digital resource by reference. `r:secureIndirect` identifies a digital resource by reference and includes a hash to bind the content reference to the actual contents of the file. These two elements are used as defined.

m1x:ProtectedResource	This element is used as defined.
r:digitalResource xenc:EncryptedData xenc:EncryptedKey m1x:resourceLocator	r:digitalResource contains the URI that points to the protected Resource, and is used as defined. xenc:EncryptedData contains the encryption data for the Resource, and is used as defined. xenc:EncryptedKey contains the encrypted symmetric key that was used for encrypting the content identified in r:digitalResource, and is used as defined. M1x:resourceLocator contains the URI that points to the location of the protected resource, and is used as defined.
r:allConditions	This element is used as defined.
r:condition	The r:allConditions element is retained in the profile, so that other conditions can be grouped together by it and used conjunctively. Used as defined.
r:validityInterval	This element is used as defined.
r:notBefore r:notAfter	r:notBefore and r:notAfter are used to specify a fixed interval of time. Used as defined.
sx:validityIntervalFloating	This element is restricted to include the following child element only: sx:duration.
sx:duration	sx:duration is used to specify the duration of a floating interval of time. Used as defined.
sx:exerciseLimit	This element is restricted to include the following child element only: sx:count.
sx:count	sx:count is used to specify the maximum number of exercises allowed. Used as defined.
sx:territory	This element includes the following child element only: sx:location and sx:domain, but with sx:location being restricted to sx:country and sx:region.
sx:location	This element is restricted to include the following child elements only: sx:country and sx:region.
sx:country sx:region	sx:country and sx:region are used to specify ISO-3166 country and region codes for identifying a country and region. Used as defined.
sx:domain	sx:domain is used to specify a network domain identified by an URI. Used as defined.
sx:validityTimeMetered	This element is restricted to include the following child element only: sx:duration and sx:quantum.
sx:duration sx:quantum	A metered interval of time is specified with the length being the value of the sx:duration and incremental being the value of sx:quantum. Used as defined.
m1x:drmSystem	This element is used as defined.
m1x:identifier	m1x:identifier identifies the class of DRM systems.

r:issuer	This element is restricted to include the following child elements only: dsig:Signature, m1x:identityHolder and r:keyHolder.
dsig:Signature m1x:identifyHolder r:keyHolder	dsig:Signature, used as defined, gives the hash of the license, the public key of the issuer, and the signature of the issuer over the license. m1x:identifyHolder provides an identity of the issuer, and is used as defined. r:keyHolder gives the public key of the issuer, and is used as defined.
m1x:serviceLocation	This element It specifies a (permission) server by its location m1x:url indicating where the server is located. It is used in the element r:serviceReference element within the m1x:seekPermission element to the location of a server from which the permission for exercising the associated right must be sought.

Insert a new Annex K as follows:

Annex K
(informative)

Example Use Case Scenarios of the Profiles

This annex contains selected use case scenarios of the profiles in Clause 10.

K.1 MAM Profile

K.1.1 Bind Rights to User's IMSI Identifier

Alice has a phone with an IMSI identifier. In this example, the device with the specified IMSI identifier is granted the right to play the stated digital resource.

```
<?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:m1x="urn:mpeg:mpeg21:2005:01-REL-M1X-NS"
xmlns:dsig="http://www.w3.org/2000/09/xmldsig#" xmlns:enc="http://www.w3.org/2001/04/xmlenc#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:mpeg:mpeg21:2005:01-REL-M1X-
NS m1x.xsd">
    <r:grant>
        <m1x:identityHolder>
            <idSystem>urn:acme:imsi</idSystem>
            <idValue>IMSI:2232111123</idValue>
        </m1x:identityHolder>
        <mx:play/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
        </r:digitalResource>
    </r:grant>
    <r:issuer>
        <r:keyHolder>
            <r:info>
                <dsig:KeyName>Rights Issuer Public Key Name</dsig:KeyName>
            </r:info>
        </r:keyHolder>
    </r:issuer>
</r:license>
```

K.1.2 Preview Rights

The rights issuer sends Alice a license that allows her to preview (play once) her favourite video clip. The license is bound to the IMSI identifier and has an exercise limit of 1.

```
<?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:m1x="urn:mpeg:mpeg21:2005:01-REL-M1X-NS"
xmlns:dsig="http://www.w3.org/2000/09/xmldsig#" xmlns:enc="http://www.w3.org/2001/04/xmlenc#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:mpeg:mpeg21:2005:01-REL-M1X-
NS m1x.xsd">
    <r:grant>
        <m1x:identityHolder>
            <idSystem>urn:acme:imsi</idSystem>
            <idValue>IMSI:2232111123</idValue>
        </m1x:identityHolder>
        <mx:play/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
        </r:digitalResource>
        <sx:exerciseLimit>
            <sx:count>1</sx:count>
        </sx:exerciseLimit>
    </r:grant>
    <r:issuer>
        <r:keyHolder>
            <r:info>
                <dsig:KeyName>Rights Issuer Public Key Name</dsig:KeyName>
            </r:info>
        </r:keyHolder>
    </r:issuer>
</r:license>
```

K.1.3 Basic Download

A rights issuer sends Alice a license that allows her to play her favorite video clip under the following conditions:

1. until a particular date and time, and
2. as long as the specified metered usage time of 2 weeks is not exceeded

In this example, the license is bound to Alice through her IMSI identifier.

```
<?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:m1x="urn:mpeg:mpeg21:2005:01-REL-M1X-NS"
xmlns:dsig="http://www.w3.org/2000/09/xmldsig#" xmlns:enc="http://www.w3.org/2001/04/xmlenc#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:mpeg:mpeg21:2005:01-REL-M1X-
NS m1x.xsd">
    <r:grant>
        <m1x:identityHolder>
            <idSystem>urn:acme:imsi</idSystem>
            <idValue>IMSI:2232111123</idValue>
        </m1x:identityHolder>
        <mx:play/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:movie:clips:hero_trailer.mpeg"/>
        </r:digitalResource>
        <r:allConditions>
            <r:validityInterval>
                <r:notAfter>2004-02-13T15:30:00</r:notAfter>
            </r:validityInterval>
            <sx:validityTimeMetered>
                <sx:duration>P14D</sx:duration>
            </sx:validityTimeMetered>
        </r:allConditions>
    </r:grant>
    <r:issuer>
        <r:keyHolder>
            <r:info>
                <dsig:KeyName>Rights Issuer Public Key Name</dsig:KeyName>
            </r:info>
        </r:keyHolder>
    </r:issuer>
</r:license>
```

K.1.4 Super-Distribution

Alice receives a protected resource (a movie clip) via a local link from her friend. Alice acquires a license from the rights issuer to allow her to play this movie clip on her phone. The rights issuer binds the license to Alice's IMSI number and provides the encrypted symmetric key to decrypt the protected resource.

Note that the phone's public key was used to encrypt the symmetric CEK and the corresponding private key is needed to decrypt the encrypted symmetric CEK.

In addition, Alice can forward the protected resource to other consumers, but those consumers must obtain their own license from the license issuer. Alice license is non-transferable because it's bound to her IMSI identifier.

```
<?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:m1x="urn:mpeg:mpeg21:2005:01-REL-M1X-NS"
xmlns:dsig="http://www.w3.org/2000/09/xmldsig#" xmlns:enc="http://www.w3.org/2001/04/xmlenc#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:mpeg:mpeg21:2005:01-REL-M1X-
NS m1x.xsd">
    <r:grant>
        <m1x:identityHolder>
            <idSystem>urn:acme:imsi</idSystem>
            <idValue>IMSI:2232111123</idValue>
        </m1x:identityHolder>
        <mx:play/>
        <m1x:protectedResource licensePartId="video">
            <r:digitalResource>
                <r:nonSecureIndirect URI="http://acme.org/myVideo"/>
            </r:digitalResource>
            <xenc:EncryptedKey>
                <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-1_5"/>
                <dsig:KeyInfo>
                    <dsig:KeyName>Phone's Public Key Name</dsig:KeyName>
                </dsig:KeyInfo>
                <xenc:CipherData>
                    <xenc:CipherValue>AQABAA==</xenc:CipherValue>
                </xenc:CipherData>
                <xenc:CarriedKeyName>Content encryption key for movie clip</xenc:CarriedKeyName>
            </xenc:EncryptedKey>
        </m1x:protectedResource>
    </r:grant>
    <r:issuer>
        <r:keyHolder>
            <r:info>
                <dsig:KeyName>Rights Issuer's Public Key Name</dsig:KeyName>
            </r:info>
        </r:keyHolder>
    </r:issuer>
</r:license>
```

K.1.5 Copying Protected Content to a Specified DRM System

Alice obtains a license from a rights issuer that allows her to play her protected content on her phone and allows her to copy the protected content to an external DRM system called ACME DRM system version 1.0. In this example, the specified DRM system gets a copy of the protected content and follows some governance rules called "AcmeCopyRules" to obtain the next set of rights for the copy. The rules require the resulting rights do not contain the `m1x:governedCopy` Right, but they include the other rights in the original license.

```
<?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:m1x="urn:mpeg:mpeg21:2005:01-REL-M1X-NS"
xmlns:dsig="http://www.w3.org/2000/09/xmldsig#" xmlns:enc="http://www.w3.org/2001/04/xmlenc#"
```

```
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:mpeg:mpeg21:2005:01-REL-M1X-
NS m1x.xsd">
    <r:grant>
        <m1x:identityHolder licensePartId="device">
            <idSystem>urn:acme:imsi</idSystem>
            <idValue>IMSI:2232111123</idValue>
        </m1x:identityHolder>
        <mx:play/>
        <m1x:protectedResource licensePartId="video">
            <r:digitalResource>
                <r:nonSecureIndirect URI="http://acme.org/myVideo"/>
            </r:digitalResource>
            <xenc:EncryptedKey>
                <xenc:EncryptionMethod Algorithm=" http://www.w3.org/2001/04/xmlenc#rsa-1_5"/>
                <dsig:KeyInfo>
                    <dsig:KeyName>Phone's Public Key Name</dsig:KeyName>
                </dsig:KeyInfo>
                <xenc:CipherData>
                    <xenc:CipherValue>AQABAA==</xenc:CipherValue>
                </xenc:CipherData>
                <xenc:CarriedKeyName>Content encryption key for movie clip</xenc:CarriedKeyName>
            </xenc:EncryptedKey>
        </m1x:protectedResource>
    </r:grant>
    <r:grant>
        <m1x:identityHolder licensePartIdRef="device"/>
        <m1x:governedCopy governanceRule="AcemCopyRules"/>
        <m1x:protectedResource licensePartIdRef="video"/>
        <m1x:drmSystem>
            <m1x:identifier>urn:ACME_DRM_SYSTEM:Version10</m1x:identifier>
        </m1x:drmSystem>
    </r:grant>
    <r:issuer>
        <r:keyHolder>
            <r:info>
                <dsig:KeyName>Rights Issuer's Public Key Name</dsig:KeyName>
            </r:info>
        </r:keyHolder>
    </r:issuer>
</r:license>
```

After the `m1x:governedCopy` right is exercised, the rights associated to the copy of the protected resource are as shown in the following license.

```
<?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:m1x="urn:mpeg:mpeg21:2005:01-REL-M1X-NS"
xmlns:dsig="http://www.w3.org/2000/09/xmldsig#" xmlns:enc="http://www.w3.org/2001/04/xmlenc#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:mpeg:mpeg21:2005:01-REL-M1X-
NS m1x.xsd">
    <r:grant>
        <m1x:identityHolder licensePartId="device">
            <idSystem>urn:acme:imsi</idSystem>
            <idValue>IMSI:2232111123</idValue>
        </m1x:identityHolder>
        <mx:play/>
        <m1x:protectedResource licensePartId="video">
            <r:digitalResource>
                <r:nonSecureIndirect URI="http://acme.org/myVideo"/>
            </r:digitalResource>
            <xenc:EncryptedKey>
                <xenc:EncryptionMethod Algorithm=" http://www.w3.org/2001/04/xmlenc#rsa-1_5"/>
                <dsig:KeyInfo>
                    <dsig:KeyName>Phone's Public Key Name</dsig:KeyName>
                </dsig:KeyInfo>
                <xenc:CipherData>
                    <xenc:CipherValue>AQABAA==</xenc:CipherValue>
                </xenc:CipherData>
                <xenc:CarriedKeyName>Content encryption key for movie clip</xenc:CarriedKeyName>
            </xenc:EncryptedKey>
        </m1x:protectedResource>
    </r:grant>
    <r:issuer>
        <r:keyHolder>
            <r:info>
                <dsig:KeyName>Rights Issuer's Public Key Name</dsig:KeyName>
            </r:info>
        </r:keyHolder>
    </r:issuer>
</r:license>
```

K.1.6 Subscription Example

This example has 2 licenses. The first license states that the holder of the named key possesses a “premier membership” subscription. In essence, this is a membership certificate to the specified subscription.

```
<?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:dsig="http://www.w3.org/2000/09/xmldsig#" xmlns:mx="urn:mpeg:mpeg21:2003:01-REL-MX-NS"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:mpeg:mpeg21:2003:01-REL-MX-NS
rel-mx-m1x-v1.xsd">
    <r:grant>
        <r:keyHolder>
            <r:info>
                <dsig:KeyName>Alice</dsig:KeyName>
            </r:info>
        </r:keyHolder>
        <r:possessProperty/>
        <sx:propertyUri definition="urn:acme:subscription:member:premier"/>
        <r:validityInterval>
            <r:notBefore>2005-02-14T00:00:00</r:notBefore>
            <r:notAfter>2005-02-24T00:00:00</r:notAfter>
        </r:validityInterval>
    </r:grant>
</r:license>
```

The second license states that premier subscribers (those who hold the membership certificate) can play any premier-subscription content. These 2 licenses are enough to allow any premier subscriber to play any marked premier-subscription content without requiring an individual license for each content file. Since this license grants rights over a collection of contents, it does not carry the content encryption key for any content.

```
<?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:dsig="http://www.w3.org/2000/09/xmldsig#" xmlns:mx="urn:mpeg:mpeg21:2003:01-REL-MX-NS"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:mpeg:mpeg21:2003:01-REL-MX-NS
rel-mx-m1x-v1.xsd">
    <r:grant>
        <r:forAll varName="subscriber">
            <r:propertyPossessor>
                <sx:propertyUri definition="urn:acme:subscription:member:premier"/>
            </r:propertyPossessor>
        </r:forAll>
        <!-- A pattern that matches any content whose URI starts with "http://acme.com/premier" -->
        <r:forAll varName="subscriptionContent">
            <r:anXmlExpression>r:nonSecureIndirect/@URI[starts-with(.,
"http://acme.com/premier")]</r:anXmlExpression>
        </r:forAll>
        <r:principal varRef="subscriber"/>
        <mx:play/>
        <r:digitalResource varRef="subscriptionContent"/>
    </r:grant>
</r:license>
```

An alternative second license can be specified to allow premier subscribers to play a *specific* content. In contrast to the previous license, this license carries the content encryption key for the specified resource. Here, it is assumed that every subscriber has obtained a subscription pair of public and private keys.

```
<?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:m1x="urn:mpeg:mpeg21:2005:01-REL-M1X-NS"
xmlns:dsig="http://www.w3.org/2000/09/xmldsig#" xmlns:enc="http://www.w3.org/2001/04/xmlenc#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:mpeg:mpeg21:2005:01-REL-M1X-
NS m1x.xsd">
    <r:grant>
        <r:forAll varName="subscriber">
            <r:propertyPossessor>
                <sx:propertyUri definition="urn:acme:subscription:member:premier"/>
            </r:propertyPossessor>
        </r:forAll>
        <r:principal varRef="subscriber"/>
        <mx:play/>
        <m1x:protectedResource>
            <r:digitalResource>
                <r:nonSecureIndirect URI="http://acme.org/myVideo"/>
            </r:digitalResource>
            <xenc:EncryptedKey>
                <xenc:EncryptionMethod Algorithm=" http://www.w3.org/2001/04/xmlenc#rsa-1_5"/>
                <dsig:KeyInfo>
                    <dsig:KeyName>Subscription Public Key Name</dsig:KeyName>
                </dsig:KeyInfo>
                <xenc:CipherData>
                    <xenc:CipherValue>AQABAA==</xenc:CipherValue>
                </xenc:CipherData>
                <xenc:CarriedKeyName>Content encryption key for movie clip</xenc:CarriedKeyName>
            </xenc:EncryptedKey>
        </m1x:protectedResource>
    </r:grant>
</r:license>
```

K.1.7 Domain Example

In this example, there are also 2 licenses, much like the subscription example, but now considering a collection of devices as an authorized domain, rather than a collection of content as a subscription. The first license states that the holder of the named key is a member of the domain by virtue of possessing the stated property. This is in essence a domain certificate for the device.

```
<?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:dsig="http://www.w3.org/2000/09/xmldsig#" xmlns:mx="urn:mpeg:mpeg21:2003:01-REL-MX-NS"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:mpeg:mpeg21:2003:01-REL-MX-NS
rel-mx-m1x-v1.xsd">
    <r:grant>
        <r:keyHolder>
            <r:info>
                <dsig:KeyName>A Domain Device</dsig:KeyName>
            </r:info>
        </r:keyHolder>
        <r:possessProperty/>
        <sx:propertyUri definition="urn:acme:domain:device"/>
        <r:validityInterval>
```

```
            <r:notBefore>2005-02-14T00:00:00</r:notBefore>
            <r:notAfter>2005-02-24T00:00:00</r:notAfter>
        </r:validityInterval>
    </r:grant>
</r:license>
```

The second license states that any device in the stated domain (meaning any device that has the stated domain certificate) can play the specified resource. These two licenses are enough to allow all domain members to play the stated content. The benefit of this two-license model is that device-specific licenses for each piece of content are not needed (although the MPEG REL does support device-specific licenses for each piece of content).

```
<?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:dsig="http://www.w3.org/2000/09/xmldsig#" xmlns:mx="urn:mpeg:mpeg21:2003:01-REL-MX-NS"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:mpeg:mpeg21:2003:01-REL-MX-NS
rel-mx-m1x-v1.xsd">
    <r:grant>
        <r:forAll varName="domainDevice">
            <r:propertyPossessor>
                <sx:propertyUri definition="urn:acme:domain:device"/>
            </r:propertyPossessor>
        </r:forAll>
        <r:principal varRef="domainDevice"/>
        <mx:play/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="http://acme.org/myVideo"/>
        </r:digitalResource>
    </r:grant>
</r:license>
```

K.1.8 Optical Media Examples

K.1.8.1 Assumptions

All of the example application scenarios for optical media described in this subclause have the following assumptions:

- Content and device: In the optical disc, there are some pre-recorded content such as several versions of the same movie (e.g., theatre release version, special edition for DVD, etc), trailers for promotion, and/or music on sound tracks. An optical disc player has a function for on-line connectivity and it can play streamed content and/or store download content from a remote server. A part of usage control information is described in a title usage file (TUF). Customized TUFs and related play lists for individual target users may be available through on-line transactions.

- Typical user1: Lisa has an optical disc player which has the function to support instant permissions. She likes movies and often uses a rental video service. She also has a portable optical disc player for her family's outdoor holidaymaking.

- Typical user2: Shun has a high-definition optical disc player which has the function to support cacheable permissions. He is a movie freak and likes to collect package media of his favourites. He always wants to get special rights related to his collection.

K.1.8.2 Example 1: Director's cut edition on theatrical release version

Lisa borrowed a movie which her friend recommended. Beyond her expectation, this movie is very exciting and she wants to watch the Director's cut version. Then she selected this version from the menu and got its playback permission. Because her player recognized that the condition for the playback right is satisfied, she begins to play the Director’s cut version.

```
<?xml version="1.0" encoding="UTF-8"?>
<r:license xmlns:r="urn:mpeg:mpeg21:2003:01-REL-R-NS" xmlns:mx="urn:mpeg:mpeg21:2003:01-REL-MX-NS"
xmlns:m1x="urn:mpeg:mpeg21:2005:01-REL-M1X-NS" xsi:schemaLocation="urn:mpeg:mpeg21:2005:01-REL-M1X-NS
m1x.xsd">
    <r:grant>
        <m1x:identityHolder licensePartId="LisaHighDefDVDPlayer">
            <idSystem>urn:acme:HD</idSystem>
            <idValue>223211112</idValue>
        </m1x:identityHolder>
        <mx:play/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:directorCutVersion"/>
        </r:digitalResource>
        <m1x:seekPermission>
            <r:serviceReference>
                <m1x:serviceLocation>
                    <m1x:url>http://www.foo.org/paymentService/payPerView</m1x:url>
                </m1x:serviceLocation>
            </r:serviceReference>
        </m1x:seekPermission>
    </r:grant>
</r:license>
```

K.1.8.3 Example 2: High definition content playback control

Shun bought the newest high definition display. Then he went to a rental shop to borrow her favourite movie. However, when he looked a menu, she realized that she had to wait high definition playback for two more months because the rental version was prohibited to play before the end of this year.

```
<?xml version="1.0" encoding="UTF-8"?>
<r:license xmlns:r="urn:mpeg:mpeg21:2003:01-REL-R-NS" xmlns:mx="urn:mpeg:mpeg21:2003:01-REL-MX-NS"
xmlns:m1x="urn:mpeg:mpeg21:2005:01-REL-M1X-NS" xsi:schemaLocation="urn:mpeg:mpeg21:2005:01-REL-M1X-NS
m1x.xsd">
    <r:grant>
        <m1x:identityHolder licensePartId="ShunHighDefDVDPlayer">
            <idSystem>urn:acme:HD</idSystem>
            <idValue>223298765</idValue>
        </m1x:identityHolder>
        <mx:play/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="http://acme.org/aFavouriteVideo"/>
        </r:digitalResource>
        <r:allConditions>
            <m1x:startCondition>
                <!-- Suppose the current date is 2005-11-20 -->
                <r:validityInterval>
                    <r:notBefore>2006-01-20T00:00:00</r:notBefore>
                </r:validityInterval>
            </m1x:startCondition>
            <m1x:outputRegulation>
                <m1x:regulation typeOfSignal="m1x:digital" qualityOfSignal="m1x:HD">ICT:1</m1x:regulation>
            </m1x:outputRegulation>
        </r:allConditions>
    </r:grant>
</r:license>
```

K.1.8.4 Example 3: Actors' commentary download

Lisa borrowed a movie in which her favourite actor appeared. Because she wanted to hear his commentary, she selected the alternative version which plays the main movie with his commentary together. Then the player connected to a remote server and downloaded the commentary.

```
<?xml version="1.0" encoding="UTF-8"?>
<r:license xmlns:r="urn:mpeg:mpeg21:2003:01-REL-R-NS" xmlns:mx="urn:mpeg:mpeg21:2003:01-REL-MX-NS"
xmlns:m1x="urn:mpeg:mpeg21:2005:01-REL-M1X-NS" xsi:schemaLocation="urn:mpeg:mpeg21:2005:01-REL-M1X-NS
m1x.xsd">
    <r:grant>
        <m1x:identityHolder licensePartId="LisaHighDefDVDPlayer">
            <idSystem>urn:acme:HD</idSystem>
            <idValue>223211112</idValue>
        </m1x:identityHolder>
        <mx:play/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="http://acme.org/aFavouriteVideo"/>
        </r:digitalResource>
        <m1x:derivationConstraint >
            <m1x:resourceInclusionList>
                <m1x:protectedResource>
                    <r:nonSecureIndirect URI="http://acme.org/commentary"/>
                </m1x:protectedResource>
            </m1x: resourceInclusionList >
        </m1x: derivationConstraint>
    </r:grant>
</r:license>
```

K.1.8.5 Example 4: Extra contents for visitors

Lisa's family had a plan to visit a theme park on next Sunday. At a book store, she looked for guide books for the theme park and found a guide book with an optical disc attached. This attached disc contained extra pre-recorded contents which can play only when the player recognizes a specific service server located in the theme park. Because her portable player can seek the service location, she selected and bought it.

NOTE For "SeekPermission" with "serviceLocation" example.

```
<?xml version="1.0" encoding="UTF-8"?>
<r:license xmlns:r="urn:mpeg:mpeg21:2003:01-REL-R-NS" xmlns:mx="urn:mpeg:mpeg21:2003:01-REL-MX-NS"
xmlns:m1x="urn:mpeg:mpeg21:2005:01-REL-M1X-NS" xsi:schemaLocation="urn:mpeg:mpeg21:2005:01-REL-M1X-NS
m1x.xsd">
    <r:grant>
        <m1x:identityHolder licensePartId="LisaHighDefDVDPlayer">
            <idSystem>urn:acme:HD</idSystem>
            <idValue>223211112</idValue>
        </m1x:identityHolder>
        <mx:play/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:themeParkVideo"/>
        </r:digitalResource>
        <m1x:seekPermission>
            <r:serviceReference>
                <m1x:serviceLocation>
                    <m1x:url>http://www.acme-theme-park.org/permissionService</m1x:url>
                </m1x:serviceLocation>
            </r:serviceReference>
        </m1x:seekPermission>
    </r:grant>
</r:license>
```

K.1.8.6 Example 5: Special version for collectors

Shun bought the final volume of his favourite TV series. The last package contained a special play list with its TUF which allows to playback a special version of the series. The usage rule requires all of the permission for individual discs as its condition. Because Shun can satisfy this condition, he can download the new play list from a remote server for the special version.

NOTE some of special version examples are listed below:

- A sequel to the story.
- A version with another ending.
- A seamless playback version of the whole story with skipping of each opening, ending, and overlap scenes of all episodes of a TV series.

```
<?xml version="1.0" encoding="UTF-8"?>
<r:license xmlns:r="urn:mpeg:mpeg21:2003:01-REL-R-NS" xmlns:mx="urn:mpeg:mpeg21:2003:01-REL-MX-NS"
xmlns:m1x="urn:mpeg:mpeg21:2005:01-REL-M1X-NS" xsi:schemaLocation="urn:mpeg:mpeg21:2005:01-REL-M1X-NS
m1x.xsd">
    <r:grant>
        <m1x:identityHolder licensePartId="ShunHighDefDVDPlayer">
            <idSystem>urn:acme:HD</idSystem>
            <idValue>223298765</idValue>
        </m1x:identityHolder>
        <mx:play/>
        <r:digitalResource>
            <r:nonSecureIndirect URI="urn:someTVSeries:specialVersion"/>
        </r:digitalResource>
        <r:allConditions>
            <m1x:seekPermission>
                <r:serviceReference>
                    <m1x:serviceLocation>
                        <m1x:url>http://www.TVStation.org/permissionService/someTVSeries/1-in-3</m1x:url>
                    </m1x:serviceLocation>
                </r:serviceReference>
            </m1x:seekPermission>
            <m1x:seekPermission>
                <r:serviceReference>
                    <m1x:serviceLocation>
                        <m1x:url>http://www.TVStation.org/permissionService/someTVSeries/2-in-3</m1x:url>
                    </m1x:serviceLocation>
                </r:serviceReference>
            </m1x:seekPermission>
            <m1x:seekPermission>
                <r:serviceReference>
                    <m1x:serviceLocation>
                        <m1x:url>http://www.TVStation.org/permissionService/someTVSeries/3-in-3</m1x:url>
                    </m1x:serviceLocation>
                </r:serviceReference>
            </m1x:seekPermission>
        </r:allConditions>
    </r:grant>
</r:license>
```

K.1.8.7 Example 6: Movie to Music Video Clip

Shun liked one of BGM in one of his favourite movies and wanted to hear the BGM with his favourite scenes in that movie. Therefore he bought a play list template for a music video clip with an associated TUF.

Using a play list editing tool, he can make a new play list by selecting music as a main content and determining the order of pieces of content from the movie as a background video.

NOTE In the TUF, there is a set of new usage rules for each piece of content. A content provider may define following usage rule;

- Whether the related piece is allowed or not allowed to use in the template,

- Mandatory piece of content to be included in the template,

- Exclusive constraint that prohibits from playing actors' speech, or

- Time order constraint between pieces such as “after”, “before” or “simultaneous”.

```
<?xml version="1.0" encoding="UTF-8"?>
<r:license xmlns:r="urn:mpeg:mpeg21:2003:01-REL-R-NS" xmlns:mx="urn:mpeg:mpeg21:2003:01-REL-MX-NS"
xmlns:m1x="urn:mpeg:mpeg21:2005:01-REL-M1X-NS" xsi:schemaLocation="urn:mpeg:mpeg21:2005:01-REL-M1X-NS
m1x.xsd">
    <r:grant>
        <m1x:identityHolder licensePartId="ShunHighDefDVDPlayer">
            <idSystem>urn:acme:HD</idSystem>
            <idValue>223298765</idValue>
        </m1x:identityHolder>
        <m1x:enlist/>
        <m1x:protectedResource>
            <r:nonSecureIndirect URI=" file:///DVDdisc/ADV_OBJ/Song1/>
            <m1x;resourceLocator> file:///DVDdisc/TM_OBJ/TM1</resourceLocator>
        </m1x:protected>
        <m1x:derivationConstraint>
            <m1x:resourceInclusionList>
                <m1x:protectedResource>
                    <r:nonSecureIndirect URI=" file:///DVDdisc/ADV_OBJ/Vo2"/>
                    <m1x;resourceLocator> file:///DVDdisc/TM_OBJ/TM2</resourceLocator>
                </m1x:protectedResource>
                <m1x:protectedResource>
                    <r:nonSecureIndirect URI="" file:///DVDdisc/ADV_OBJ/Vo3"/>
                    <m1x;resourceLocator> file:///DVDdisc/TM_OBJ/TM3</resourceLocator>
                </m1x:protectedResource>
            </m1x:resourceInclusionList>
        </m1x:derivationConstraint >
    </r:grant>
</r:license>
```