

**INTERNATIONAL ORGANISATION FOR STANDARDISATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO/IEC JTC1/SC29/WG11
CODING OF MOVING PICTURES AND AUDIO**

**ISO/IEC JTC1/SC29/WG11
MPEG2011/M20126
March 2011, Geneva, Switzerland**

Source CONVERGENCE project (FP7-257123)
Status Input contribution
Title Input contribution for DID extension
Authors Teresa Andrade (maria.andrade@inescporto.pt)
Helder Castro (hcastro@inescporto.pt)
Edoardo Radica (edoardo@cedeo.net)
Giuseppe Tropea (giuseppe.tropea@cnit.it)

1	INTRODUCTION	2
2	CONVERGENCE.....	2
2.1	Project Context	2
2.1.1	Motivation	2
2.1.2	The Versatile Digital Item Concept	2
2.1.3	Framework.....	3
2.2	Use Cases and the Need for Inter-DI Relationships	4
2.2.1	Collaborative Cultural Content Enrichment	4
2.2.2	Augmented e-Learning	5
2.2.3	Classification and Association of Real World Objects	5
2.2.4	Interconnected Animal Health Records	6
3	MPEG-21 DI'S RELATIONAL SHORTCOMINGS.....	6
4	NEW MECHANISMS PROPOSED	7
4.1	Introduction	7
4.2	Addition of Dedicated Elements.....	8
4.2.1	InterDIRelationship Element	8
4.2.2	Potential Inter-DI Relationships Classification Scheme	9
4.2.3	Inter-DI Relationships Declaration Examples	11
4.3	Employment of RDF/OWL Carrying Descriptors	13
4.3.1	Relationship Declaring Descriptors	13
4.3.2	Inter-DI Relationship Ontology	13
4.3.3	Inter-DI Relationships Declaration Example.....	15
5	CONCLUSION	16
6	REFERENCES AND RELEVANT LITERATURE	17

1 Introduction

This document proposes a mechanism to support an extension to the MPEG-21 DID and DI model to enable the interoperable declaration of relationships between Digital Items.

This contribution was spurred by the work done in the context of IST CONVERGENCE project [1]. Based on the definition and exploration of real world use cases, the project concluded that the means to adequately express inter Digital Item (DI) relationships is missing from the MPEG-21 DID standard and that adding the means to support such relationship would add value to the MPEG-21 DID standard.

The remainder of this proposal is organized as follows. Section 2 provides an overview of the CONVERGENCE's use cases from where the need for the proposed extension derives. Section 3 presents a brief overview of MPEG-21 DID and explains its current limitations in supporting the identified requirements. Section 4 analyses different alternatives to extend the standard.

2 CONVERGENCE

2.1 Project Context

2.1.1 *Motivation*

The Internet was originally conceived as a Community of Hosts that cooperate to support the exchange of unstructured information between its nodes. Today, by contrast, it is becoming a collection of things, of services, of media items and of people. In these "new Internets", the key elements are no longer "hosts" but data and services (or content). In other words, what we are observing is a shift from "host-centric networking" to "content-centric" or "data-centric" networking. This shift imposes new requirements on middleware, and on the underlying networking functionality.

Current solutions addressing these issues are proprietary, non-interoperable, restricted to specific classes of information, and lacking in key functionality required by providers and consumers of digital information.

Within the outlined context, the strategic goal of CONVERGENCE is to address these new demands, through a novel framework based on a content-based, publish-subscribe service model, which complements and enhances the current Internet architecture.

2.1.2 *The Versatile Digital Item Concept*

One of the key enablers of today's media revolution has been the emergence of broadly accepted multimedia standards – in particular the standards produced by the MPEG community. Such standards are centred on "classical" needs of the media industry and digital entertainment. Today, however, the once clear distinctions between media products and other forms of digital products, or between providers and users of digital information are increasingly blurred. There is an effective convergence between the requirements of media users, and those of other consumers and producers of digital information.

Several of these needs are addressed by existing MPEG standards. MPEG-21 already defines standard ways of providing meta-information and standard ways of describing the content and structure of complex "Digital Items". The CONVERGENCE project is driven by the idea of extending the ability to manage and trade digital objects to a

broader range of digital objects, including descriptors for Real World Objects (RWO), services and people. To encompass this new extended class of digital objects, CONVERGENCE defines the Versatile Digital Item (VDI).

The VDI is as an extension of the MPEG-21 Digital Item that acts as the fundamental unit of transaction and distribution in the CONVERGENCE framework. A VDI is a general purpose container of information about media, real world objects, services and people.

Like MPEG-21 DIs, VDIs shall bind meta-information (describing the content and structure of the item) and resources (other VDIs, audio, images, video, text, descriptors of RWOs, descriptors of People etc.). As such, a VDI “document” may be equated to a DID, and the requirements of the former are therefore translatable into the latter.

2.1.3 Framework

CONVERGENCE will integrate the VDI concept with proposals from the networking community on strategies for overcoming the limitation imposed by the current Internet host-centric architecture. The end result will be a novel content-centric, architecture based on a publish/subscribe service model.

Figure 1 depicts the planned system's functional levels (Application, Content and Network). The functionality of each functional level are realized by means of the functional blocks, named CONVERGENCE Network (CoNet), CONVERGENCE Middleware (CoMid), Community Dictionary Services (CDS), and CONVERGENCE Applications.

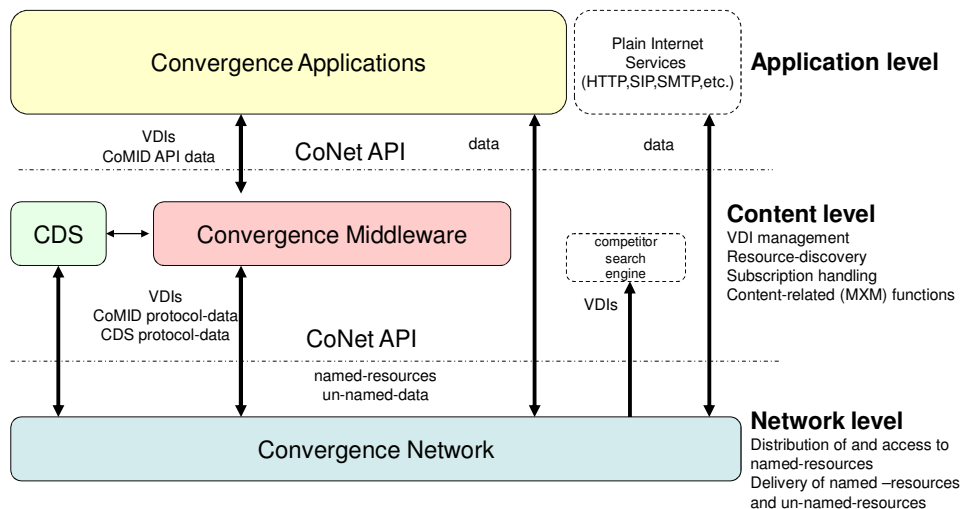


Figure 1 – CONVERGENCE Functional Levels and Technologies

The CONVERGENCE system is formed by a set of networked devices that implement the functional levels described above. Two kinds of CONVERGENCE devices exist: nodes and peers. A CONVERGENCE peer supports both CoMid and CoNet functionalities while a CONVERGENCE node only supports the CoNet functionalities.

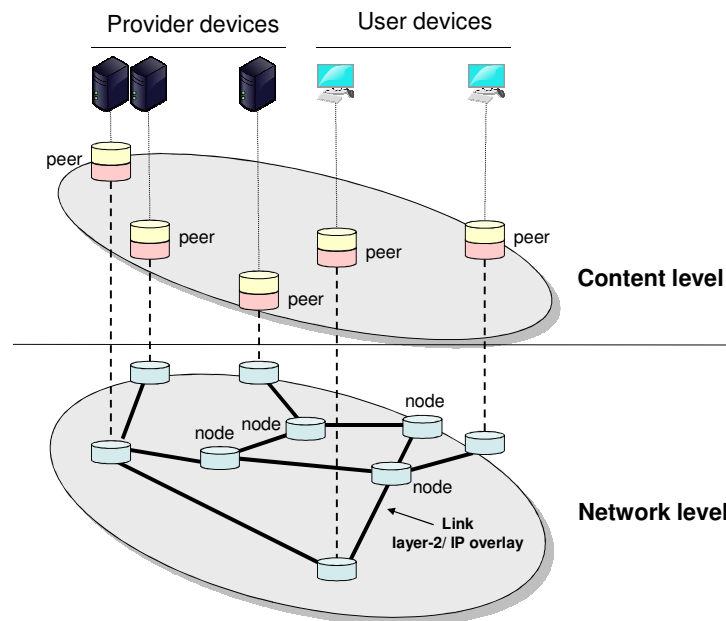


Figure 2 – Sketch of CONVERGENCE System Structure

As depicted in Figure 2, the CONVERGENCE tissue is formed by a set of peer and nodes connected to each other. Network level functionalities are accomplished in a distributed way by the whole set of CONVERGENCE nodes. Content level functionalities (and related security) are accomplished in a distributed way by the CONVERGENCE peers.

In practical terms, information providers will employ CONVERGENCE applications to publish VDIs onto the network. Information consumers will employ CONVERGENCE applications to subscribe and subsequently retrieve both specific VDIs (e.g. the repair manual for a piece of equipment) and VDIs satisfying a given search criteria (e.g. all VDIs describing concerts by a certain musician).

2.2 Use Cases and the Need for Inter-DI Relationships

Here below, excerpts of some use cases developed within CONVERGENCE are provided. These use cases refer to different applications areas and highlight the benefits of using VDIs to perform the transaction and distribution of resources in a number of diverse situations with varying requirements. In particular, they enable to pinpoint the need for additional functionality to be supported by VDIs as an extension to that offered by the MPEG-21 DI model.

2.2.1 Collaborative Cultural Content Enrichment

An important part of Simon's and his associates' audiovisual documentation on Occitan language and culture is, naturally, in Occitan. This is a barrier for people who want to use the material for teaching in an international environment.

To overcome this problem, Simon selects a part of his audiovisual documentation, produces the corresponding VDI and publishes it on his web site

allowing (and expecting) interested bilingual users to produce the associated French (or other) language written or spoken translations.

Carla and Jim download the VDI and use it to produce their own partial translations into Italian and English. Once Simon declares his acceptance of Carla's and Jim's translations, the VDIs corresponding to the produced translations are published on Simon's video site.

In this use case, the VDIs created by Carla and Jim, bearing the translations, clearly have a logical relationship with Simon's original VDI. The explicit and unambiguous declaration of that relationship in terms of an "enrichment" of the original VDI, would enable the automatic linking between the two objects so that, regardless of their collocation or separation, users would be able to "travel" between such VDIs in a seamless, direct and automated way, without any ambiguous steps in between. The possibility for users to seamlessly obtain the translation VDI while performing a search for Simon's audiovisual Occitan documentation (possibly using a keyword or semantic based query), would certainly be an added-value both for the users and well as for Simon, who would see his potential audience increasing.

2.2.2 Augmented e-Learning

Professor Tamara Watson publishes a VDI containing the material she lectures in each class, that is, in each lecturing episode. Each such VDI consists of a video and synchronized slides.

Students are free to subscribe to Watson's content.

Tom Mansfield, a computer science student who is attending the course, is automatically notified about the release of any new lecture episodes. Downloaded episodes are also automatically updated when modified, and thus are always up to date.

Tom uses the VDI browser to watch the latest episode and comes up with a doubt pertaining to some lectured subject. He produces a comment to the subject (a point in the video or in the slides, he has doubts about) and makes it public. The CONVERGENCE system then pushes the comment to all students who have subscribed to the same episode. They start a group discussion and work out a solution for Tom's problem.

The VDI created by Tom maintains a clear logical relationship with the lecture VDI, in this case a commenting relationship. For the system to diffuse Tom's comment in an automated way to all users which subscribed to the lectures' VDIs, there should be some mechanism that would declare and preserve its association (and the nature of the association) with the lecture VDI, in a automated and seamless way. This mechanism should be included in the comment VDI and should explicitly declare that its content maintains a commenting relationship with the lecture VDI. This way, by analyzing the comment VDI it is possible to automatically determine its purpose and to distribute it to all interested parties, without requiring any action from the users' side. This feature allows creating much richer and engaging e-Learning scenarios, with clear benefits for the participants.

2.2.3 Classification and Association of Real World Objects

John, an electronic manufacturer, is releasing a new type of Blue-Ray player to the market. Being CONVERGENCE-enabled, John creates and publishes a new VDI for this type of product, in order to publicize it. Said VDI represents the abstract entity "new Blue-Ray player model". It contains a number of resources covering a broad range of technical and non-technical information (e.g. manufacturer, wiring diagrams, parts details, users manuals, firmware details...).

Meanwhile, Mark, a big retailer who deals with John frequently, finds out about John's new model and purchases a number of them.

Alice goes to Mark's store and buys a new Blue-Ray player. After she has paid, the salesman generates a new VDI, a specific Blue-Ray player VDI for the individual player she has bought and gives it to her. This new VDI inherits relevant information from the abstract VDI (issued from John) for the particular model she has bought.

Alice's VDI (that describes her individual Blue-ray player) clearly maintains a relationship with the original and abstract VDI produced and distributed by John (which describes the abstract type of Blue-Ray player to which Alice's player belongs). As such, Alice's VDI "instantiates" John's VDI, in the sense that it declares an instance of a thing whose type is declared by John's VDI.

For this relationship to be easily, clearly and immediately perceivable from the VDIs in an automated fashion, it should be possible for Alice's VDI to explicitly declare the relationship that it participates in together with John's VDI. The advantages of this seamless relationship identification is, for example, the possibility of automatically notifying buyers of a given product of possible malfunctioning detected in one of the items and of explaining how to solve the problem.

2.2.4 Interconnected Animal Health Records

Ronnie owns a cow farm that is certified as organic by BIOPRODUCTS, and the animals in the farm have to be vaccinated. James is a veterinarian; he is hired by the local authorities to perform the vaccination. At the farm, with the help of his mobile phone, he reads the animals' ear tags and remotely retrieves the VDIs with their corresponding health records.

He proceeds with the vaccination, and emits (publishes with the appropriate services), for each animal, a vaccination certification VDI, which is linked to the animal's health record VDI and also to the VDI containing James' own veterinary certificate.

A few days later, Rory, the representative of BIOPRODUCTS, arrives at the farm, to inspect it and renew its certificate. He reads out the animals' VDIs, and through them he is able to retrieve linked VDIs emitted by the veterinarian, checking if vaccination and, if desired, veterinary certificate of James, are both ok. He then updates the animals' VDIs with the certification from BIOPRODUCTS.

The VDIs emitted by the veterinary and the VDIs bearing animal's health record are logically related. The former extend the latter. Furthermore, a relationship can also be identified between the VDIs emitted by the veterinary and the veterinary's own certification VDI, given that the former mentions the latter.

For this animal health system to operate smoothly while employing VDIs, thus providing automatically all the information related to the animal's vaccination to the animal's owner, or to another veterinary or any other authorized entity, it must be possible to express all mentioned inter-DI relationships within the VDIs themselves.

3 MPEG-21 DI's Relational Shortcomings

The MPEG-21 DID standard provides the tools for the declaration of Digital Items. These tools enable the declaration of a myriad of different inner structures for such Items and also the declaration of some relationship between them.

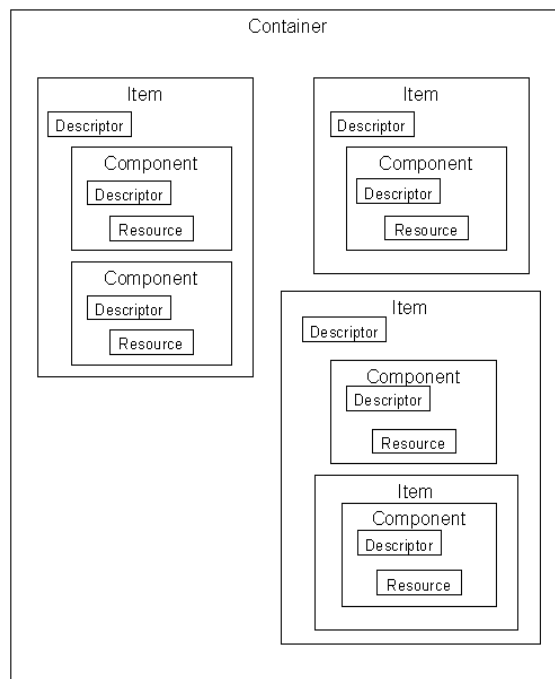


Figure 3 – Example Digital Item Declaration model

Still, the declaration of such structures and, especially, such relationships, is performed in a rigid and implicit manner that consists of placing (XML) elements within elements according to a strict linear hierarchy (as presented in Figure 3).

This scheme does not allow for the expression of a multitude of non-hierarchical, unambiguous and multidimensional relationships, which indeed occur between Digital Items in the real world, as highlighted through the description of CONVERGENCE use cases. However, as indicated, it would be useful to be able to capture these relationships within the MPEG-21 DID.

The possibility to explicitly declare Inter-DI relationships, within MPEG-21 DID, would be a simple and powerful way to enable establishment of a rich semantic inter-webbing of the global DID tissue.

In our socially interconnected time, this would greatly empower application and systems which employ MPEG-21.

4 NEW MECHANISMS PROPOSED

4.1 Introduction

A semantic relationship between digital items constitutes a logical connection between them. For instance, if DI *A* is a correction to DI *B*, then it may be said that DI *A* “*corrects*” DI *B*, that is, that DI *A* is logically bound to DI *B* through a logical relationship of “*correction*”.

In such a relationship, the participating DIs may play two different roles:

- The active role – this role is played by the DI which “performs” the relationship. In the previous example, VDI *A* “performs” the correction of VDI *B*, and, as such, plays the active role in the “correction” relationship.
- The passive or object role – this role is played by the DI upon which the relationship is “performed”. In the previous example, VDI *B* has a correction “performed” upon it by VDI *A*, and, as such, plays the object role in the “correction” relationship.

The expression of such relationships will enable the employment of MPEG-21 DID’s standard for the webbing of rich, multi-dimensional DI structures, presenting a greater social usefulness and overall value for the users.

This contribution therefore sustains that MPEG-21 DIDs should support the explicit declaration of semantic relationships between DIs. We envision two main alternatives to enable such a declaration:

- The addition, to the standard, of a dedicated inter-DI relationship-describing element. This element would perform the explicit declaration of inter-DI relationships, which are themselves defined in a standard manner through a classification scheme;
- The addition to individual DIDs, of Descriptors carrying RDF/OWL metadata tailored for the description of the inter-DI relationships in which the (Descriptor’s) enclosing Item takes part.

4.2 Addition of Dedicated Elements

4.2.1 *InterDIRelationship* Element

An *InterDIRelationship* element (and its substructure), is proposed to be added to the standard to provide the means to express Inter-DI relationships. This element should be carried within *Statement* elements, which are inside *Descriptor* elements. The latter are in turn children of *Item* or *Container* elements.

The *InterDIRelationship* element’s internal semantics pertains to the DI represented by its ancestor *Item* or *Container* element. That is, they describe one or more relationships in which the DI, (represented by the ancestor *Item* or *Container* element), participates and identify the other DIs involved in those relationships.

An *InterDIRelationship* element thus binds relationship pertaining data to its ancestor *Item* or *Container* element.

An *InterDIRelationship* may further have any attribute from other namespaces. Such attributes provide additional information about the relationship represented by the *InterDIRelationship* element.

Within the *InterDIRelationship* element, the *DIRelationship* element, (as presented in Table 1), carries the identification of the inter-DI relationship in question.

Furthermore, the *InterDIRelationship* element contains a sequence of:

- Either *ActorDI* elements – each *ActorDI* element identifies a DI which plays the active role in a relationship (with the enclosing DI), of the type identified by the sibling *DIRelationship* element.

- Or *ObjectDI* elements – each *ObjectDI* element identifies a DI which plays the passive role in a relationship (with the enclosing DI), of the type identified by the sibling *DIRelationship* element.

Diagram			
Attributes	Name	Type	Description
	id	ID	A unique ID value.
Source	<pre> <xs:element name="InterDIRelationship" type="InterDIRelationshipType"/> <xs:complexType name="InterDIRelationshipType"> <xs:sequence> <xs:element name="DIRelationship" type="xs:anyURI"/> <xs:choice> <xs:element name="ActorDI" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element ref="dii:RelatedIdentifier"/> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="ObjectDI" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element ref="dii:RelatedIdentifier"/> </xs:sequence> </xs:complexType> </xs:element> </xs:choice> </xs:sequence> </xs:complexType> </pre>		

Table 1 — *InterDIRelationship* element syntax

If a sequence of *ActorDI* elements is present (and thus *ObjectDI* is missing) that means that the DI declaring the relationship plays the passive role in the relationship (the role represented by the missing *ObjectDI*). On the other hand, if a sequence of *ObjectDI* elements is present (and thus *ActorDI* is missing), that means that the DI declaring the relationship plays the active role in the relationship (the role represented by the missing *ActorDI*).

4.2.2 Potential Inter-DI Relationships Classification Scheme

The specific relationships declared within the *DIRelationship* element (child of *InterDIRelationship*), should be defined within a standard Inter-DI Relationship classification scheme.

That classification scheme must contain a core set of global inter-DI relationships, which may be extended by any application that employs MPEG-21 DID.

Figure 4 presents an example of a possible such classification scheme.

```
<?xml version="1.0" encoding="UTF-8"?>
<ClassificationScheme uri="urn:mpeg:mpeg21:didl:cs:InterDIRelationshipCS">
  <Term termID="1">
    <Name xml:lang="en">correction</Name>
    <Definition xml:lang="en">
      A relationship where a correcting DI supplies information
      whose purpose is to correct other information present in a
      corrected DI
    </Definition>
  </Term>
  <Term termID="2">
    <Name xml:lang="en">responding</Name>
    <Definition xml:lang="en">
      Lists the relationships where a DI exists somehow in
      response to another DI
    </Definition>
    <Term termID="2.1">
      <Name xml:lang="en">commenting</Name>
      <Definition xml:lang="en">
        A relationship where a commenting DI supplies
        information whose purpose is to comment upon other
        information present in a commented DI
      </Definition>
    </Term>
    <Term termID="2.2">
      <Name xml:lang="en">evaluation</Name>
      <Definition xml:lang="en">
        A relationship where an evaluating DI supplies
        information whose purpose is to provide an evaluation
        of some kind over an evaluated DI
      </Definition>
    </Term>
  </Term>
  <Term termID="3">
    <Name xml:lang="en">extension</Name>
    <Definition xml:lang="en">
      Lists the relationships where a DI logically extends
      another DI
    </Definition>
    <Term termID="3.1">
      <Name xml:lang="en">enrichment</Name>
      <Definition xml:lang="en">
        A relationship where an enriching DI supplies
        information whose purpose is to complement/enrich the
        information contained in another DI which is thus
        being enriched
      </Definition>
    </Term>
    <Term termID="3.2">
      <Name xml:lang="en">completation</Name>
    </Term>
  </Term>
</ClassificationScheme>
```

```

        <Definition xml:lang="en">
            A relationship where a completing DI supplies
            information whose purpose is to deliver complementary
            information to another DI which is missing it
        </Definition>
    </Term>
</Term>
<Term termID="4">
    <Name xml:lang="en">sequence</Name>
    <Definition xml:lang="en">
        Lists the relationships pertaining to DI sequences, where
        one DI logically follows or precedes another DI
    </Definition>
    <Term termID="4.1">
        <Name xml:lang="en">precedence</Name>
        <Definition xml:lang="en">
            A relationship where a DI (which participates in a DI
            sequence with at least another DI) logically precedes
            another DI
        </Definition>
    </Term>
    <Term termID="4.2">
        <Name xml:lang="en">succession</Name>
        <Definition xml:lang="en">
            A relationship where a DI (which participates in a DI
            sequence with at least another DI) logically
            succeeds another DI
        </Definition>
    </Term>
</Term>
</ClassificationScheme>

```

Figure 4 – Inter-DI Relationships Classification Scheme Example

4.2.3 Inter-DI Relationships Declaration Examples

Figure 5 presents an example of a DI declaration where the declared DI is identified as **conv:di:002**, and is declared to maintain a relationship (where it plays the active role) of correction (identified as urn:mpeg:mpeg21:didl:cs:InterDIRelationshipCS:1), with another DI identified as **conve:di:001**. That is, the contents of DI **conv:di:002** constitute a correction to the contents of DI **conve:di:001**.

```

<DIDL xmlns="urn:mpeg:mpeg21:2002:02-DIDL-NS"
  xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001"
  xmlns:foo="http://www.bar.org/title-schema"
  xmlns:dii="urn:mpeg:mpeg21:2002:01-DII-NS"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Container>
    <Item>
      <Descriptor>
        <Statement mimeType="text/xml">
          <dii:Identifier>
            conv:di:002
          </dii:Identifier>
        </Statement>
      </Descriptor>
      <Descriptor>
        <Statement mimeType="text/xml">

```

```

        <InterDIRelationship>
            <DIRelationship>
                urn:mpeg:mpeg21:didl:cs:InterDIRelationshipCS:1
            </DIRelationship>
            <ObjectDI>
                <dii:RelatedIdentifier>
                    conv:di:001
                </dii:RelatedIdentifier>
            </ObjectDI>
        </InterDIRelationship>
    </Statement>
</Descriptor>
<Descriptor>
    <Statement mimeType="text/xml">
    </Statement>
</Descriptor>
<Component>
    <Resource ref="movie.mpg" mimeType="video/mpeg"/>
</Component>
</Item>
</Container>
</DIDL>

```

Figure 5 – *InterDIRelationship* Element Example A

Figure 6 presents an example of a DI declaration where the declared DI is identified as **conv:di:004**, and is declared to maintain a relationship (where it plays the passive role) of precedence (identified as **urn:mpeg:mpeg21:didl:cs:InterDIRelationshipCS:4.1**), with another DI identified as **conve:di:003**. That is, DI **conv:di:003** logically precedes DI **conv:di:004**.

```

<DIDL xmlns="urn:mpeg:mpeg21:2002:02-DIDL-NS"
  xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001"
  xmlns:foo="http://www.bar.org/title-schema"
  xmlns:dii="urn:mpeg:mpeg21:2002:01-DII-NS"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Container>
    <Item>
      <Descriptor>
        <Statement mimeType="text/xml">
          <dii:Identifier>
            conv:di:004
          </dii:Identifier>
        </Statement>
      </Descriptor>
      <Descriptor>
        <Statement mimeType="text/xml">
          <InterDIRelationship>
            <DIRelationship>
              urn:mpeg:mpeg21:didl:cs:InterDIRelationshipCS:4.1
            </DIRelationship>
            <ActorDI>
              <dii:RelatedIdentifier>
                conve:di:003
              </dii:RelatedIdentifier>
            </ActorDI>
          </InterDIRelationship>
        </Statement>
      </Descriptor>
      <Descriptor>
        <Statement mimeType="text/xml">

```

```

        </Statement>
      </Descriptor>
    </Component>
    <Resource ref="movie.mpg" mimeType="video/mpeg"/>
  </Component>
</Item>
</Container>
</DIDL>

```

Figure 6 – *InterDlRelationship* Element Example B

4.3 Employment of RDF/OWL Carrying Descriptors

4.3.1 Relationship Declaring Descriptors

The expression of inter-DI relationships may be carried out through the introduction of inter-DI relationship declaring *Descriptors*.

Each such *Descriptor* binds, to its parent element (Digital Item), a portion of RDF/OWL metadata that describes the relationships in which that DI participates with other DIs.

The specific format of the mentioned RDF/OWL metadata should be in accordance with a defined global ontology.

Said ontology may also be extended on an application dependent basis.

4.3.2 Inter-DI Relationship Ontology

Figure 7 presents an example of a possible Inter-DI Relationship classification ontology, which contains the same relationships as those defined in the Classification scheme of section 4.2.2.

```

<?xml version="1.0"?>

<!DOCTYPE rdf:RDF [
  <!ENTITY owl "http://www.w3.org/2002/07/owl#" >
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >
  <!ENTITY owl2xml "http://www.w3.org/2006/12/owl2-xml#" >
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
  <!ENTITY Ontology1299523590765
"http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl#" >
]>

<rdf:RDF
xmlns="http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl#"
xml:base="http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl"
xmlns:owl2xml="http://www.w3.org/2006/12/owl2-xml#"
xmlns:Ontology1299523590765="http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl#"
xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:owl="http://www.w3.org/2002/07/owl#">
<owl:Ontology rdf:about=""/>

  <!-- Object Properties -->

```

```

    <!--
http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl#commenting
-->
    <owl:ObjectProperty rdf:about="#commenting">
        <rdfs:subPropertyOf rdf:resource="#responding"/>
    </owl:ObjectProperty>

    <!--
http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl#completion
-->
    <owl:ObjectProperty rdf:about="#completion">
        <rdfs:subPropertyOf rdf:resource="#extension"/>
    </owl:ObjectProperty>

    <!--
http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl#correction
-->
    <owl:ObjectProperty rdf:about="#correction">
        <rdfs:subPropertyOf rdf:resource="#interDIRelationship"/>
    </owl:ObjectProperty>

    <!--
http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl#enrichment
-->
    <owl:ObjectProperty rdf:about="#enrichment">
        <rdfs:subPropertyOf rdf:resource="#extension"/>
    </owl:ObjectProperty>

    <!--
http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl#evaluation
-->
    <owl:ObjectProperty rdf:about="#evaluation">
        <rdfs:subPropertyOf rdf:resource="#responding"/>
    </owl:ObjectProperty>

    <!--
http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl#extension
-->
    <owl:ObjectProperty rdf:about="#extension">
        <rdfs:subPropertyOf rdf:resource="#interDIRelationship"/>
    </owl:ObjectProperty>

    <!--
http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl#interDIRelationship
-->
    <owl:ObjectProperty rdf:about="#interDIRelationship">
        <rdfs:domain rdf:resource="#DigitalItem"/>
        <rdfs:range rdf:resource="#DigitalItem"/>
    </owl:ObjectProperty>

    <!--
http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl#precedence
-->
    <owl:ObjectProperty rdf:about="#precedence">
        <rdfs:subPropertyOf rdf:resource="#sequence"/>
    </owl:ObjectProperty>

```

```

<!--
http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl#responding
-->
<owl:ObjectProperty rdf:about="#responding">
  <rdfs:subPropertyOf rdf:resource="#interDIRelationship"/>
</owl:ObjectProperty>

<!--
http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl#sequence
-->
<owl:ObjectProperty rdf:about="#sequence">
  <rdfs:subPropertyOf rdf:resource="#interDIRelationship"/>
</owl:ObjectProperty>

<!--
http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl#succession
-->
<owl:ObjectProperty rdf:about="#succession">
  <rdfs:subPropertyOf rdf:resource="#sequence"/>
</owl:ObjectProperty>

<!-- Classes -->

<!--
http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl#DigitalItem
-->
<owl:Class rdf:about="#DigitalItem"/>
</rdf:RDF>

```

Figure 7 – Potential *InterDIRelationship* OWL Ontology fragment

4.3.3 *Inter-DI Relationships Declaration Example*

Figure 8 presents an example of a DI declaration where the declared DI is identified as **conv:di:005**, and is declared to maintain a relationship of *enrichment*, with another DI identified as **conv:di:006**. That is, DI **conv:di:005** logically enriches DI **conv:di:006**.

```

<?xml version="1.0" encoding="UTF-8"?>
<DIDL xmlns="urn:mpeg:mpeg21:2002:02-DIDL-NS"
  xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001"
  xmlns:foo="http://www.bar.org/title-schema"
  xmlns:dii="urn:mpeg:mpeg21:2002:01-DII-NS"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Container>
    <Item>
      <Descriptor>
        <Statement mimeType="text/xml">
          <dii:Identifier>
            conv:di:005
          </dii:Identifier>
        </Statement>
      </Descriptor>
      <Descriptor>
        <Statement mimeType="text/xml">
          <rdf:RDF
            xmlns="http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl#" xml:base="
            http://www.semanticweb.org/ontologies/2011/2/Ontology1299523590765.owl"
            xmlns:owl2xml="http://www.w3.org/2006/12/owl2-xml#"

```

```

xmlns:Ontology1299523590765="http://www.semanticweb.org/ontologies/2011/2/Ontology1299523
590765.owl#"
xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:owl="http://www.w3.org/2002/07/owl#"
    <owl:Ontology rdf:about="" />

    <DigitalItem rdf:about="#conv:di:005">
        <rdf:type rdf:resource="&owl;Thing" />
        <enrichment rdf:resource="#conv:di:006" />
    </DigitalItem>
    <DigitalItem rdf:about="#conv:di:006">
        <rdf:type rdf:resource="&owl;Thing" />
    </DigitalItem>

</rdf:RDF>
</Statement>
</Descriptor>
<Descriptor>
    <Statement mimeType="text/xml">
        </Statement>
    </Descriptor>
    <Component>
        <Resource ref="movie.mpg" mimeType="video/mpeg" />
    </Component>
</Item>
</Container>
</DIDL>

```

Figure 8 – Inter-DI Relationship Declaring Descriptor Example

5 Conclusion

It is proposed that the use cases be reviewed, requirements discussed and a solution adopted between the two presented.

Although the final syntax of the proposed approach is left open, in the following some notes and remarks are made regarding the two approaches, trying to summarize the immediately visible advantages and disadvantages of each.

Addition of Dedicated Elements PROs

- not being triples, as it is the case with RDF statements, it is easy to have multiple subjects or objects of the relationship
- distinguishing between object and subject of the relationship might help querying for more precise results and more intuitively model the semantic domain, instead of using directed relationships
- might prove to be more efficiently parsed compared to existing RDF parsers, and more easily developed in the context of MPEG standards
- allows a MPEG-21 self-contained expression of the relationships, i.e. there is no need for additional external standards

Addition of Dedicated Elements CONS

- not based on existing standards
- formulating queries, like SPARQL does on RDF statements, might not be easy
- cannot be immediately connected to existing ontologies, i.e. relationships cannot be expressed in Terms of already existing ontologies

- introduction of a custom "Classification Scheme" necessary, instead of using OWL

Employment of RDF/OWL PROs

- parsers, tools, query languages exist
- a wide set of ontologies exist covering different semantic domains

Employment of RDF/OWL CONs

- cumbersome, somewhat inefficient
- multiple subjects/objects must be split into separate statements

This brief analysis seems to show that the RDF/OWL approach is the easiest to integrate and test, given the availability of off-the-shelf software able to manipulate such XML structures. Moreover it seamlessly allows exploiting a wide range of semantic models, covering different domains, which have been developed in OWL format already. Furthermore, considering that usage of RDF/OWL constructs is taking place in other parts of the MPEG standards (Media Value Chain Ontology, Contracts, ...) we think the RDF/OWL approach is the most mature one.

6 References and Relevant Literature

- [1] CONVERGENCE, <http://www.ict-convergence.eu/>
- [2] Chrisa Tsinaraki, Panagiotis Polydoros, and Stavros Christodoulakis; Interoperability Support between MPEG-7/21 and OWL in DS-MIRF
- [3] Using MPEG-21 DIDL to Represent Complex Digital Objects in the Los Alamos National Laboratory Digital Library;
http://www.dlib.org/dlib/november03/bekaert/11bekaert.html#Appendix_A
- [4] Victor Rodriguez-Doncel, Jaime Delgado; A Media Value Chain Ontology for MPEG-21, IEEE