Technical Report, No. TR-2003302, Dept. of Computer Science and Business Informatics, University of Vienna, June, 2003

CULTOS: Towards a World-Wide Digital Collection of Exchangeable Units of Multimedia Content for Intertextual Studies

Karin Schellner, Utz Westermann, Sonja Zillner, Wolfgang Klas

Dept. of Computer Science and Business Informatics
University of Vienna, Austria
E-mail: {karin.schellner, gerd-utz.westermann, sonja.zillner, wolfgang.klas}@univie.ac.at

Abstract

It is the aim of the CULTOS project to provide researchers in intertextual studies with a collaborative multimedia platform for the authoring, management, search, exchange, and presentation of Intertextual Threads (ITTs), knowledge structures that interrelate and compare cultural artefacts. By means of the CULTOS platform, researchers will be able to create a world-wide collection of multimedia-enhanced ITTs comparing cultural artefacts from different personal and cultural backgrounds. This constitutes a valuable contribution for comparative studies and cultural heritage. In this paper, we propose and formally define Enhanced Multimedia Meta Objects (Emmos) as a new means for representing multimedia content. Emmos are unique in that they combine three different aspects of multimedia content: the individual media objects making up the content, semantic relationships between those media objects, and functionality on the content. With Emmos, we obtain an adequate means for the representation of multimedia-enriched ITTs as the foundation for the world-wide distributed collection of ITTs envisioned by CULTOS.

1. Introduction

CULTOS is an EU-funded project that started in September 2001 and is carried out by 11 partners from different EU-countries, Israel, and Estonia.[1] CULTOS addresses the need of researchers in the domain of intertextual studies for an integrated view on individual and culture-dependent perceptions of interrelationships between cultural artefacts. For that purpose, an Internet-based multimedia collaboration platform is being developed that provides tools and infrastructures for authoring, managing, retrieving, exchanging, and presenting so-called Intertextual Threads (ITTs) [4] – knowledge structures that semantically interrelate and compare cultural artefacts such as literature, artworks, movies, etc. On the basis of the CULTOS platform, the community of intertextual studies will be able to create and exchange pieces of a world-wide collection of semantically connected and multimedia-enriched artefacts incorporating the communities' different cultural backgrounds. Hence, the project constitutes an ambitious endeavor concerning the preservation of European cultural heritage.

In this paper, we present first major results of our research in CULTOS. More precisely, we motivate our work in CULTOS by giving an introduction to the concept of ITTs and illustrating the benefits a distributed multimedia platform for the collaborative authoring and dissemination of ITTs on the Internet has for researchers in intertextual studies. As the heart of this platform, we propose and formally define Enhanced Multimedia Meta Objects (Emmos) – a novel means for representing multimedia content in a tradeable fashion that suits the needs of ITTs. Emmos indivisibly encompass the individual media objects of which multimedia content consists, the semantic relationships between the media, and functionality on the content. We perform a study of related standards and approaches showing the novelty of Emmos. We provide a sound, formal specification of the Emmo model and outline the implementation of a basic management component for Emmos as the cornerstone of the CULTOS platform.

The remainder of the paper is organized as follows: Section 2 gives an introduction to Intertextual Threads and explains the basic idea of Emmos. Section 3 takes a look on related work. Section 4 provides a formal definition of the Emmo model. Section 5 briefly covers our implementation an Emmo management component. Section 6 concludes this paper and gives an outlook to current and future work.

[1] See http://www.cultos.org for further details on the project.

2. From ITTs to Emmos

In this section, we provide a better understanding of Intertextual Threads and the aims of the CULTOS project. We begin with an illustration of Intertextual Threads and their basic characteristics (2.1). We then indicate the prospects of moving Intertextual Threads to the Internet (2.2). Finally, we give an outline of the basic ideas behind the means by which we intend to provide Internet-capable Intertextual Threads in the CULTOS project, namely Enhanced Multimedia Meta Objects (2.3).

2.1. Basic Characteristics

A central task of researchers in intertextual studies is to discover the relationships between pieces of literature and other works of art thereby elaborating Intertexual Threads (ITTs). ITTs can be represented with graphical structures that may take a variety of forms, ranging from spiders over centipedes to associative maps as shown in Figure 1.

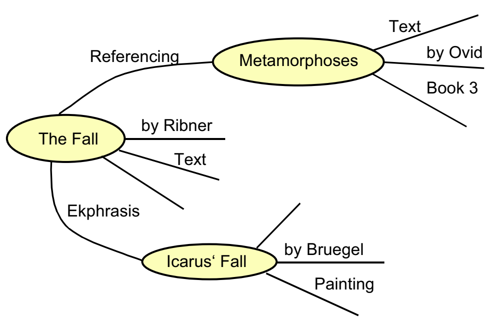

Figure 1. Simple Intertextual Thread

The example ITT depicted in the figure highlights several relationships of the poem "The Fall" by Tuvia Ribner to other works of art. It states that the poem makes reference to the 3rd book of Ovid's "Metamorphoses" and that the poem is an ekphrasis of the painting "Icarus' Fall" of the famous Dutch painter Bruegel.

When looking at the ITT, well-known techniques from the domain of knowledge engineering like conceptual graphs and semantic nets immediately come to mind. Indeed, the depicted graphical representation of the ITT bears a strong resemblance to such techniques, though it lacks their formal rigidity. However, the complexity of ITTs should not be underestimated. ITTs commonly make use of constructs that are very challenging from the perspective of knowledge representation, such as *encapsulation* and *reification of statements*.

Encapsulation is intrinsic to ITTs because intertextual studies are not exact sciences. Certainly, the cultural and personal context of a researcher will affect the kinds of relationships between pieces of literature and works of art that are discovered and of importance to the researcher. This inevitably results in differences and even contradictions between different ITTs created by two different researchers on the same subject. As there thus cannot be a global "truth", every ITT is a "truth" in its own right that has to be protected by an encapsulating impenetrable boundary. Moreover, differences on a certain subject are highly interesting facts for researchers in intertextual studies. Consequently, ITTs themselves can be relevant subjects of discourse and thus be contained as first-class artefacts within other ITTs.

Reification of statements is yet another demanding construct frequently occurring within ITTs. Since experts in intertextual studies extensively base their position on the position of other researchers, statements about statements are common practice within ITTs. Typically, reification is not just a one-step process: statements about already reified statements are no rarity.

We illustrate these points with a more complex ITT that is given by Figure 2.

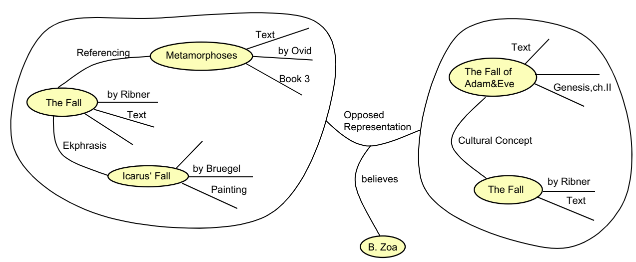

Figure 2. Complex Intertextual Thread

Here, two ITTs that manifest two very different points of view on Ribner's poem are depicted: while it has been important for the author of the ITT to the left which we already know from Figure 1 to see "The Fall" in the context of other works of art, the author of the ITT to the right stresses the relationship of the poem to the religious concept of the "Fall of Adam and Eve" in the "Genesis". For a third author, the difference between both viewpoints has been motivation enough to oppose both ITTs with a statement within a third ITT. However, it is also expressed by reifying that statement that this is only the belief of the researcher B. Zoa. Other researchers might very well consider both ITTs not as opposing but rather as complementing viewpoints.

2.2. Bringing ITTs to the Internet

Up to date, ITTs have been treated within traditional books and articles. However, the Internet has become an increasingly accepted tool in intertextual studies during the recent years [3]. At the same time, the technical community has begun to recognize the central importance of describing the semantic aspects of content for the further development of the Internet towards a Semantic Web (e.g., [5, 11]). Indeed, bringing both worlds together bears several appealing prospects:

Representing ITTs and the relationships between works of art that they establish in a machine-processable way allows to build *digital collections* of ITTs. These collections could offer tools for *searching*, *browsing*, and *querying* ITTs. Such collections would be valuable resources for researchers in intertextual studies.

Moreover, an ITT could be furnished with *digital* media that constitute manifestations of the pieces of literature and works of art covered by the ITT. The ability to consume these media while browsing an ITT will certainly enhance the comprehension of the ITT and the semantic relationships within.

Additionally, digital collections on the Internet could support the *collaborative* authoring of ITTs. This would allow different researchers with different cultural backgrounds to work together and establish interesting links between works of art. Also, valuable insights on opposing views of different researchers on a subject could be gained.

Furthermore, an ITT in machine-processable form could be augmented with *functionality*. For instance, an ITT could know how to render itself as a SMIL presentation or PDF document and could be capable of automatically clearing rights on digital media before viewing.

Finally, ITTs in machine-processable form could be serialized into an exchangeable format and thereby become *tradeable*. Hence, ITTs could be exchanged between distributed collections. Going a step further, one could imagine to refrain from building centralized collections of ITTs. Instead, each researcher could locally maintain a collection of ITTs which he or she could share and trade with other researchers via a peer-to-peer network.

2.3. Enhanced Multimedia Meta Objects

In order to let the promises of Internet-based ITTs become reality, it has been the task of our group in the CULTOS project to create a suitable foundation for their representation. Starting out from an abstract idea originally formulated by [25], we have developed such a mechanism for the representation of ITTs: Enhanced Multimedia Meta Objects (Emmos).

An Emmo is a self-contained unit of multimedia content that encompasses three aspects, which we would like to illustrate using Figure 3 that depicts a sketch of an Emmo representing the ITT of Figure 1:

1. *The media aspect:* An Emmo aggregates the media objects of which the multimedia content consists. In the figure, we see that the depicted Emmo contains the PDF document "The Fall.pdf", the text document

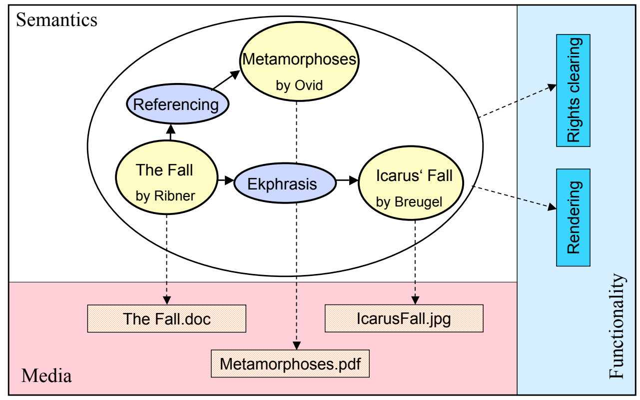

Figure 3. Example Emmo

"Metamorphoses.txt", and the JPEG image "Icarus Fall.jpeg". Containment of media objects can be realized either by inclusion, i.e., the raw media data is embedded within an Emmo, or by reference via an URI, in cases where embedding media data is not feasible.

2. *The semantic aspect:* An Emmo further encapsulates semantic associations between its contained media objects by means of a graph-based model similar to conceptual graphs. Hence, an Emmo constitutes a unit of expert knowledge concerning the multimedia content. In the example figure, it is stated that the media objects contained within the Emmo are digital manifestations of Ribner's poem "The Fall", Ovid's "Metamorphoses", and Bruegel's painting "Icarus' Fall". Also, the interpretation of the author of the original ITT has been remodeled: "The Fall" makes a reference to "Metamorphoses" and constitutes an ekphrasis of "Icarus' Fall". The model used for semantic associations is expressive: it is, e.g., possible to establish references to other Emmos and to reify associations.

3. *The functional aspect:* An Emmo offers operations for dealing with its content which applications can invoke. In the figure, the depicted Emmo is associated with two operations: one for rendering the Emmo which might return a presentation of the ITT in different formats, such as SMIL and SVG. Another operation is provided to clear the rights for the media objects contained in the Emmo before rendering, for example by performing a credit card transaction or by displaying terms of usage.

Emmos have further desirable characteristics. They can be *serialized* into a bundle that completely encompasses all three aspects. Thus, an Emmo is *transferable* in its entirety between different Emmo providers, including its contained media objects, semantic associations between these objects, and functionality. Moreover, *versioning support* has been a central design objective: all the constituents of an Emmo can be versioned, thereby paving the way for the distributed, *collaborative construction* of Emmos.

3. Related Standards and Approaches

The fundamental idea underlying the concept of Emmos is that an Emmo is an object unifying three different aspects of multimedia content, namely the media aspect, the semantic aspect, and the functional aspect. In the following, we fortify our claim that this idea is unique by comparing the Emmo idea with other approaches to the representation of multimedia content.

Interrelating basic media objects like single images and videos to form *multi*media content is the task of multimedia document models. Recently, several standards for multimedia document models have emerged [6], such as HTML [24], XHTML+SMIL [22], HyTime [15], MHEG-5 [13], MPEG-4 BIFS and XMT [23], SMIL [1], and SVG [12]. Multimedia document models can be regarded as composite media formats that model the presentation of multimedia content by arranging basic media objects according to temporal, spatial, and interaction relationships. They thus mainly address the media aspect of multimedia content. Compared to Emmos, however, multimedia document models neither interrelate multimedia content according to semantic aspects nor do they allow to provide functionality on the content. They rely on external applications like presentation engines for reasonable content processing.

In parallel to research concerning the Semantic Web, a variety of standards have appeared that can be used to model multimedia content by describing the information it conveys on a semantic level, such as RDF [19, 7], Topic Maps [16], MPEG-7 (especially MPEG-7's graph tools for the description of content semantics [14]), and Conceptual Graphs [17]. These standards clearly cover the semantic aspect of multimedia content. As they also offer means to address media objects within a description, they undoubtedly refer to the media aspect of multimedia content as well. Compared to Emmos, however, these approaches do not provide functionality on multimedia content. They rely on external software like database and knowledge base technology, search engines, user agents, etc. for the processing of content descriptions. Furthermore, media descriptions and the media objects described are separate entities – potentially scattered around different places on the Internet, created and maintained by different and unrelated authorites not necessarily aware of each other and not necessarily synchronized – whereas Emmos combine media objects and their semantic relationships into a single indivisible unit.

There exist several approaches that represent multimedia content by means of objects. Enterprise Media Beans (EMBs) [2] extend the Enterprise Java Beans (EJBs) architecture [21] with predefined entity beans for the representation of basic media objects within enterprise applications. These come with rudimental access functionality but can be extended with arbitrary functionality using the inheritance mechanisms available to all EJBs. Though addressing the media and functional aspects of content, EMBs in comparison to Emmos are mainly concerned with single media content and not with *multi*media content. Furthermore, EMBs do not offer any dedicated support for the semantic aspect of content.

Adlets [9] are objects that represent individual (not necessarily multimedia) documents. Adlets support a fixed set of predefined functionality which enables them to advertise themselves to other Adlets. They are thus content representations that address the media as well as the functional aspect. Different from Emmos, however, the functionality supported by Adlets is limited to advertisement and there is no explicit modeling of the semantic aspect.

Tele-Action Objects (TAOs) [8] are object representations of multimedia content that encapsulate the basic media objects of which the content consists and interlink these objects with associations. Though TAOs thus address the media aspect of multimedia content in a way similar to Emmos, they do not adequately cover the semantic aspect of multimedia content: only a fixed set of 5 association types is supported mainly concerned with temporal and spatial relationships for presentation purposes. TAOs can further be augmented with functionality. Such functionality is, in contrast to the functionality of Emmos, automatically invoked as the result of system events and not explicitly invoked by applications.

Distributed Active Relationships [10] define an object model based on the Warwick Framework [18]. In the model, Digital Objects (DOs), which are interlinked with each other by semantic relationships, act as containers of metadata describing multimedia content. DOs thus do not address the media aspect of multimedia content but focus on the semantic aspect. The links between containers can be supplemented with arbitrary functionality. As a consequence, DOs take account of the functional aspect as well. Different from Emmos, however, the functionality is not explicitly invoked by applications but implicitly whenever an application traverses a link between two DOs.

4. The Emmo Model

This section describes and formally defines the Emmo model for the representation of multimedia content. It illustrates how the model can be used to build and represent multimedia-enhanced ITTs.

We begin by introducing the concept of *entities* which constitute an abstract notion subsuming the different constituents of the Emmo model (4.1). We then define the four concrete specializations of entities, namely *logical media parts* representing media objects, *ontology objects* representing concepts of an ontology, *associations* modeling binary relationships between entities, and *Emmos* themselves which are aggregations of semantically related entities (4.2 – 4.5).

4.1. Entity

Before we start with a formal definition of the abstract notion of entities, we clarify some basic symbols required for the definitions to follow:

Definition 1 (Symbols) *Let* Γ *denote the set of all logical media parts,* Θ *the set of all ontology objects,* Λ *the set of all associations,* Σ *the set of all Emmos, and* $\Omega = \Gamma \cup \Theta \cup \Lambda \cup \Sigma$ *the set of all entities.*

Further, let $\mathcal{MS}$ *be the set of all media selectors,* $\mathcal{MP}$ *the set of all media profiles,* $\mathcal{OP}$ *the set of all operations.*

Finally, let $\mathbb{UUID}$ *be the set of all universal unique identifiers,* $\mathbb{STR}$ *the set of all strings,* $\mathbb{OBJ}$ *the set of all objects,* $\mathbb{URI}$ *the set of all uniform resource identifiers,* $\mathbb{RMD}$ *the set of all raw media data, and* $\mathbb{FUN}$ *the set of all functions.*

Based on these common symbols, the definition of entities is formulated below:

Definition 2 (Entity) *An entity* $w \in \Omega$ *is a thirteen-tuple* $w = (o_w, n_w, k_w, s_w, t_w, T_w, A_w, C_w, N_w, P_w, S_w, F_w, O_w)$*, where* $o_w \in \mathbb{UUID}$ *denotes the unique object identifier (OID) of* w*,* $n_w \in \mathbb{STR}$ *the name of* w*,* $k_w \in \{"lmp", "ont", "asso", "emm"\}$ *the kind of* w*,* $s_w \in \Omega \cup \{\varepsilon\}$ *the source and* $t_w \in \Omega \cup \{\varepsilon\}$ *the target entity of* w *with* $\varepsilon \notin \Omega$ *stating that such an entity is undefined,* $A_w \subseteq \Theta \times \mathbb{OBJ}$ *the attribute values,* $T_w \subseteq \Theta$ *the types,* $C_w \subseteq \mathcal{MS} \times \mathcal{MP}$ *the connectors,* $N_w \subseteq \Omega$ *the nodes,* $P_w \subseteq \Omega$ *the predecessors,* $S_w \subseteq \Omega$ *the successors,* $F_w \subseteq \mathbb{STR} \times \mathbb{OBJ}$ *the features, and* $O_w \subseteq \mathcal{OP}$ *the operations of* w*. The following constraints hold for all entities:*

$$\forall w_1, w_2 \in \Omega : o_{w_1} = o_{w_2} \longrightarrow w_1 = w_2 \qquad (1)$$

$$\forall w, v \in \Omega : v \in P_w \vee v \in S_w \longrightarrow k_w = k_v \qquad (2)$$

According to the definition, an entity w is globally and uniquely identified by its OID o_w as ensured by Constraint (1). Since we have chosen o_w to be a universal unique identifier (UUID) [20], OIDs can easily be generated even in a distributed scenario like the CULTOS project. As UUIDs are not really useful to humans, an entity can be augmented with a human readable name n_w which is a string. The kind k_w serves to identify whether an entity is either a logical media part, an ontology object, an association, or an Emmo.

An entity w may further have an arbitrary number of types T_w. Types are concepts taken from an ontology, so for instance, an entity might be an instantiation of the concepts ”book” and ”ancient text”; another might instantiate the concept ”painting”, etc. By attaching types, an entity gets meaning and is classified in an application-dependent ontology. In the Emmo model, types are represented as ontology objects and thus constitute entities themselves.

An entity can additionally be described by an arbitrary number of attribute values A_w. Attribute values are simple attribute-value pairs with the attribute being a concept of an application-dependent ontology (similar to types represented by an ontology object in the Emmo model) and the value being an arbitrary object. With attribute values, it is for instance possible to state that a photograph has been taken at February 17th 2003 by attaching the attribute value “date=02/17/2003” to the entity representing that photograph in the Emmo model. The attribute “date” would be an ontology object and the value “02/17/2003” would probably be a date value. The rationale behind representing attributes as concepts of an ontology and not just as simple string identifiers is that this allows to express constraints on the usage of attributes within the ontology, e.g., which entity types attributes are applicable to.

As already mentioned, the CULTOS project intends to develop a distributed platform allowing researchers in intertextual studies to work collaboratively on ITTs. In such an environment, different versions of their work will accrue not only due to the temporal evolution of a researcher’s viewpoints but also due to concurrent work of different reseachers on the same ITTs. Since different versions of ITTs are highly interesting facts to researchers, it is important to be able to trace these versions and to interrelate them within other ITTs. The Emmo model takes account of this need for versioning by allowing any entity w to have an arbitrary number of preceding versions P_w and succeeding versions S_w. A version of w is again an entity of the same kind k_w, as expressed by Constraint (2). Treating an entity’s versions as entities on their own has several benefits: on the one hand, entities constituting versions of other entities have their own globally unique OID. Hence, different versions concurrently derived from one and the same entity at different sites can easily be distinguished without synchronization effort. On the other hand, different versions of an entity can be interrelated just like any other entities allowing to establish comparative relationships between entity versions as desired in intertexual studies.

The features F_w of an entity w represent a fixed set of primitive attribute-value pairs. They have been included in the Emmo model as it might be necessary to augment entities with further attributes, e.g., timestamps or status information, in an implementation of the model.

The remaining elements and sets given by the definition – the source and target entities s_w and t_w, the connectors C_w, the nodes N_w, the operations O_w – are only relevant for certain kinds of entities. Therefore, we defer their explanation to the subsections to follow as they become relevant.

4.2. Logical Media Part

Logical media parts are entities serving to represent the media objects or parts of media objects of which multimedia content consists at a logical level within the Emmo model. When modeling a multimedia-enhanced ITT as an Emmo, logical media parts address the cultural artefacts that are subject of discourse within the ITT, for example, pieces of literature, films, paintings, etc. In order to relief authors from the burden of having digital representations of the artefacts to be treated at hand before they can start building an ITT, special care has been taken to decouple logical media parts from any existing physical representation. In fact, one can talk about Bruegel's painting "Icarus' Fall" and find intertextual relationships to other (art)works without owning a JPEG image showing that painting.

However, if an author focuses on the difference between, e.g., a concert of "Beethoven's 9th Symphony" as seen on television and the corresponding radio broadcast, the television and radio broadcasts will become two distinct media objects on a logical level and thus have to be represented by two different logical media parts. If nothing of this kind has to be expressed, a single logical media part will suffice for representing "Beethoven's 9th Symphony".

Definition 3 formally introduces logical media parts:

Definition 3 (Logical media part) *A logical media part $l \in \Gamma$ is an entity with $k_l = "lmp" \wedge s_l = t_l = \varepsilon \wedge N_l = O_l = \emptyset$.*

It is important that the definition does not restrict the set of connectors C_l of a logical media part l, which has been defined to exist for all entities in Definition 2, to an empty set: logical media parts not only model media objects at a logical level but are additionally able to maintain connections to media data representing these objects. Thereby, logical media parts provide the media aspect of multimedia content represented with the Emmo model.

Connectors (see Definition 2) consist of a media profile and a media selector. Media profiles, in accordance to the media tool set of MPEG-7 [14], represent media data. A media profile combines low-level metadata describing the media data, e.g., the storage format, with its storage locations – media instances in MPEG-7 terminology. A media instance can either address the location of media data by means of an URI or it may directly embed the media data. The ability to embed media data allows to combine media data and multimedia content described with the Emmo model based on these media into single, indivisible units.

Definition 4 formally captures media profiles and media instances in the Emmo model. Note that we abstain from providing any further details on available descriptive metadata for media profiles for reasons of space.

Definition 4 (Media profile) *A media profile $mp = (I_{mp}, M_{mp}) \in \mathcal{M}P$ is described by its media instances $I_{mp} \subseteq \mathbb{URI} \cup \mathbb{RMD}$ and its metadata $M_{mp} \subseteq \mathbb{STR} \times \mathbb{OBJ}$.*

Media selectors contained in connectors along with media profiles can address parts of the media data represented by the profile according to textual, spatial, and temporal criteria. For example, it should be possible to address a scene in a digital video starting from second 10 and lasting until second 30 without having to extract that scene and to put it into a separate file using a video editing tool.

Definition 5 introduces media selectors. Again we abstain from providing exact details on the kinds of selectors available and the parameters they take for space reasons.

Definition 5 (Media selector) *A media selector $ms = (k_{ms},\ P_{ms}) \in \mathcal{M}S$ is described by its kind $k_{ms} \in \{"spatial", "textual", "temporal", \ldots\}$ and by its parameters $P_{ms} \subseteq \mathbb{STR} \times \mathbb{OBJ}$.*

Example 1 shows how the three cultural artefacts occurring in the sketch of an Emmo in Figure 3 that covers the ITT of Figure 1 can be represented as individual logical media parts in the Emmo model. In the example, the logical media parts have been labeled l_1, l_2, and l_3. The connector of the logical media part l_2 references the upper left corner of the JPEG image file located at the URI "http://www.here.com/Icarus Fall.jpeg" which is expressed by the media profile mp_2 in combination with the spatial selector ms_2.

Example 1

$$
\begin{aligned}
l_1 =& ("a3564", "The\,Fall", "lmp", \varepsilon, \varepsilon, \{o_2\},\\
& \{o_1, "Ribner"\}, \{(ms_1, mp_1)\}, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset)\\
l_2 =& ("a7655", "Icarus'\,Fall", "lmp", \varepsilon, \varepsilon, \{o_3\},\\
& \{o_1, "Bruegel"\}, \{(ms_2, mp_2)\}, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset)\\
l_3 =& ("b4567", "Metamorphoses", "lmp", \varepsilon, \varepsilon, \{o_2\},\\
& \{o_1, "Ovid"\}, \{(ms_3, mp_3)\}, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset)\\
mp_2 =& (\{"http://www.here.com/Icarus\,Fall.jpeg"\},\\
& \{\ldots, (Format, jpeg), \ldots\})\\
ms_2 =& ("spatial", \{(topleft, (0,0)), (bottomright, (50,\\
& 50))\})
\end{aligned}
$$

4.3. Ontology Object

Ontology objects are the kind of entities that represent concepts of an ontology. As already explained, ontology objects among others serve to designate the types of entities or the attributes of attribute values attached to entities.

In the CULTOS project, the experts in intertextual studies have defined an ontology featuring the concepts necessary to represent ITTs within the Emmo model. As we have not developed an ontology language for the Emmo model yet, we follow the pragmatic approach of defining the concepts of the CULTOS ontology in an external ontology language such as RDF Schema [7] and letting the ontology objects reference these concepts. The reference to a concept defined in the external language can encoded within the features F_o of an ontology object o.

Definition 6 formally introduces ontology objects:

Definition 6 (Ontology object) *An ontology object $o \in \Theta$ is an entity with $k_o = "ont" \wedge s_o = t_o = \varepsilon \wedge C_o = N_o = O_o = \emptyset$.*

Example 2 illustrates ontology objects again using the sketch of an Emmo of Figure 3. The ontology objects o_4 and o_5 represent the types of the two associations contained in the figure, i.e., "ekphrasis" and "referencing". The ontology objects o_1, o_2 and o_3 model the types of the three logical media parts depicted although these are not explicitly addressed in the figure. The ontology objects o_6 and o_7 finally represent the designators of the operations the sketched Emmo offers (these will be explained later in conjunction with Emmos).

Example 2

$$o_1 =("c3456", "author", "ont", \varepsilon, \varepsilon, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset)$$
$$o_2 =("c4516", "text", "ont", \varepsilon, \varepsilon, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset)$$
$$o_3 =("c1162", "painting", "ont", \varepsilon, \varepsilon, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset)$$
$$o_4 =("c2356", "ekphrasis", "ont", \varepsilon, \varepsilon, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset)$$
$$o_5 =("c5627", "referencing", "ont", \varepsilon, \varepsilon, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset)$$
$$o_6 =("c4111", "rendering", "ont", \varepsilon, \varepsilon, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset)$$
$$o_7 =("c3336", "rights\,clearing", "ont", \varepsilon, \varepsilon, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset)$$

4.4. Association

Associations represent binary directed semantic relationships between entities. Thus, they provide the semantic aspect of multimedia content represented on the basis of the Emmo model. In the CULTOS project, they are in particular used to model the intertextual relationships between cultural artefacts within ITTs. Since association are "first-class" entities, they can take part in associations as well facilitating the reification of statements in the Emmo model. As we have explained before, expressing statements about statements is a very essential part of the work of experts in intertextual studies when analyzing literature and building their ITTs.

Definition 7 formally describes associations:

Definition 7 (Association) *An association $a \in \Lambda$ is an entity with $k_a = "asso" \wedge s_a \neq \varepsilon \wedge t_a \neq \varepsilon \wedge C_a = N_a = O_a = \emptyset \wedge |T_a| = 1$.*

According to the definition, the kind of semantic relationship represented by an association is defined by the association's type which is – like the types of other entities – an ontology object representing a concept taken from an ontology. Different from other entities, however, an association is only allowed to have one type as it can represent only a single kind of relationship.

Each association specifies exactly one source and one target entity s_a and t_a, and thus establishes a directed binary relationship between those two entities.

Example 3 shows the representation of the two associations given in the example Emmo of Figure 3.

Example 3

$$a_1 =("g7490", "Asso1", "asso", l_1, l_2, \{o_4\}, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset)$$
$$a_2 =("w4399", "Asso2", "asso", l_1, l_3, \{o_5\}, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset)$$

4.5. Emmo

The Emmo is the core component of our model. It is a container that groups arbitrary entities into a single unit. An Emmo can thus address the media and semantic aspects of multimedia content by aggregating media data (i.e., logical media parts) with semantic data (i.e., associations and ontology objects). The functional aspect of multimedia content can be addressed as well by augmenting the Emmo with arbitrary operations that process the content.

In CULTOS, Emmos act as the containers carrying multimedia-enhanced ITTs. In such a container, the cultural artefacts covered by an ITT are captured as logical media parts, media data digitally representing these artefacts are attached to the media parts via connectors, the intertextual relationships are modeled by means of associations, and concepts from the domain of intertextual studies are covered by ontology objects. The possibility to attach operations to an Emmo is exploited, among others, to provide ITTs with the ability to render themselves as SMIL and SVG presentations.

Since Emmos are “first-class” entities, Emmos can be contained within other Emmos just like any other entity. As a consequence, a structure of hierarchically-nested Emmos can be established. With regard to the representation of ITTs by the means of Emmos, this is of particular advantage: it is possible to interrelate two different Emmos representing two different ITTs with two different points of view on a subject within an Emmo representing a third ITT. In that manner, contradictions and relevant differences between both viewpoints can be expressed which is important for intertextual studies.

Definition 8 formally captures Emmos:

Definition 8 (Emmo) *An Emmo* $e \in \Sigma$ *is an entity with* $k_e = "emm"$*, and* $s_e = t_e = \varepsilon \wedge C_e = \emptyset$*, such that*

$$\forall x \in N_e : \quad k_x = "asso" \longrightarrow \{s_x, t_x\} \subseteq N_e \qquad (3)$$

As shown in the formal description above, an Emmo e constitutes a container of other entities because its set of nodes N_e is not restricted to an empty set, as it is the case with the other kinds of entities in the Emmo model. These contained entities form a connected graph structure when they become interlinked by associations within the Emmo e.

Only entities belonging to the Emmo's nodes can be specified as source or target entity of an association (Constraint (3)). In this way, it is guaranteed that established relationships are fully contained in an Emmo.

Definition 8 unveils a further difference between Emmos and other kinds of entites: an Emmo is more powerful in that it can have operations attached, because its set of operations O_e is not necessarily empty. In the Emmo model, an operation is basically a tuple combining an ontology object acting as the operation's designator with the operation's implementation, which can be any mathematical function. It is the intention of this modeling to achieve a high flexibility by allowing to attach arbitrarily complex operations to Emmos. In a concrete implementation of the model, an operation could be realized as a function in the underlying programming language with the full expressiveness of that language at disposal. An operation could then reference this function by means of a function pointer. We have modeled operation designators as ontology objects to be able to express constraints on operations within ontologies, e.g., for which types of Emmos an operation is available.

Definition 9 formally defines the notion of operations:

Definition 9 (Operation) *An operation* $op = (d_{op}, i_{op}) \in \mathbb{OP}$ *is described by its designator* $d_{op} \in \Theta$ *and its implementation* $i_{op} \in \mathbb{FUN}$.

To conclude the formal definition of the Emmo model, Example 4 assembles the Emmo e sketched in Figure 3 representing the ITT of Figure 1 from the entities of the other examples. The functions f_{render} and f_{rights} implement the “rendering” and “rights clearing” operations that are attached to e. For example, f_{render} could be a mathematical function that takes an Emmo as its input and transforms it to an appropriate SMIL presentation, i.e., to a string that follows the SMIL syntax.

Example 4

$$\begin{aligned} e = &("f4672", "ExampleEmmo", "emm", \varepsilon, \varepsilon, \emptyset, \emptyset, \emptyset, \emptyset, \\ &\{l_1, l_2, l_3, a_1, a_2\}, \emptyset, \emptyset, \emptyset, \{(o_6, f_{render}), (o_7, f_{rights})\}) \end{aligned}$$

5. Implementation

Having formally developed the Emmo model in the previous section, we now give a brief overview of the cornerstones of the CULTOS platform which implements that model, the so-called *Emmo Containers*. An Emmo Container is a management component for Emmos which provides a space where Emmos live. The idea is that at the site of each participant in the project which produces or consumes Emmos an Emmo Container is instantiated. Thereby, the infrastructure for a distributed collection of multimedia-enriched ITTs is established.

The core functionality offered by an Emmo Container is the persistent storage of Emmos. The container offers an API with which applications can traverse, access, and manipulate the structure of a stored Emmo in a fine-grained manner, i.e., the entities of which it consists, the associations between them, entity versions, the media profiles associated with logical media parts, etc. The API further allows to invoke and execute an Emmo's operations.

Moreover, an Emmo Container offers import/export functionality which facilitates the trading of Emmos between different Emmo Containers. On the one hand, an Emmo can be exported to a bundle which includes all of the Emmo's constituents: its entities, the raw media data associated with logical media parts among these entities, and the functionality of the Emmo. To flexibly suit the needs of different applications, several export variants are supported: Emmos can be exported with or without having media data stuffed into the bundle, Emmos contained within an exported Emmo can be recursively exported to the bundle as well, all predecessor and successor versions of exported entities can be added to the bundle.

On the other hand, an Emmo Container is able to import a bundle containing an exported Emmo. The import is aware of the fact that in a collaborative, distributed environment entities might have been changed concurrently in different Emmo Containers. Therefore, the import checks whether those entities contained in the bundle which are already stored within the container have been modified in a conflicting manner. Conflicts are then resolved on the basis of a timestamp protocol.

As the implementation environment for Emmo Containers, we have opted for Java in conjunction with the object-oriented database management system (ODBMS) ObjectStore. The choice for Java lies close at hand since Java bytecode is platform independent. This allows to implement an Emmo's operations as individual Java classes whose bytecode can be packed into a bundle and transferred between different Emmo Containers potentially running on heterogeneous platforms when an Emmo is exported. The choice for ObjectStore was not only driven by the fact that ODBMSs are well-suited for the storage of complex structures like Emmos; also, ObjectStore is scalable from a small-footprint, in-process database to a full-fledged database server without requiring applications to change code. Consequently, we are able to operate small-scale Emmo Containers directly at the locations of the Emmo providers in the project as well as large-scale Emmo Containers at dedicated servers. This gives us a lot of flexibility in building the distributed CULTOS platform.

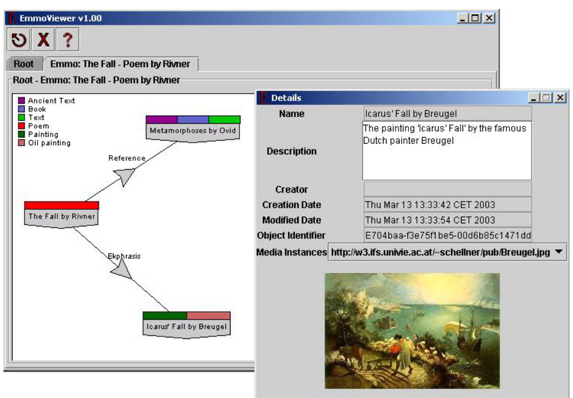

Figure 4. Screenshot of the Emmo viewer

The implementation also comes with basic administration tools, one of which is a simple graphical Emmo viewer. Figure 4 depicts a screenshot of the viewer showing the contents of the example Emmo of Figure 3 which represents the ITT of Figure 1. The logical media parts are given as pentagons which employ color codes to their top to denote their respective types. The associations between the logical media parts are represented with labeled edges. To the right, a dialog provides further detail on the logical media part representing Bruegel's painting "The Fall": among others, it displays a JPEG image acting as a media instance for a media profile associated with the logical media part.

6. Conclusion

In this paper, we have introduced the CULTOS project and its aim of providing a distributed infrastructure for the exchange and collaborative construction of multimedia-enhanced Intertextual Threads. We have proposed the unique idea of Emmos, tradeable units of multimedia content encompassing the media aspect, the semantic aspect, and the functional aspect of the content, as a suitable representation for Intertextual Threads. We have provided a formal model for Emmos and briefly outlined the implementation of an Emmo container on the basis of the model.

We are developing the Emmo approach in a variety of directions. We are formalizing a query algebra for Emmos that allows to declaratively traverse an Emmo's structure and to invoke an Emmo's operations. We are further developing an ontology language on the basis of the Emmo model. We are planning to extend the implementation of the Emmo container with support for both the query algebra and the ontology language. Moreover, we are currently wrapping the container as a peer-to-peer service facilitating distributed search, retrieval, and exchange of Emmos. Finally, we are exploring the applicability of the Emmo approach to other application scenarios such as e-Learning and the handling of security and copyright issues.

References

[1] J. Ayars, D. Bulterman, A. Cohen, et al. Synchronized Multimedia Integration Language (SMIL 2.0). W3C Recommendation, World Wide Web Consortium (W3C), August 2001.

[2] S. Baumeister. Enterprise Media Beans^{TM} Specification. Public Draft Version 1.0, IBM Corporation, March 2002.

[3] W. Behrendt, Z. Ben-Porat, and S. Reich. Organizing Multimedia Inter-textual Knowledge: New Tasks, Challenges and Technologies. In *Proc. of the Joint International Conference of the Association for Literary and Linguistic Computing and the Association for Computers and the Humanities*, Tübingen, Germany, 2002.

[4] M. Benari, Z. Ben-Porat, W. Behrendt, S. Reich, K. Schellner, and S. Stoye. Organising the Knowledge of Arts and Experts for Hypermedia Presentation. In *Proc. of the Conference of Electronic Imaging and the Visual Arts*, Florence, Italy, May 2002.

[5] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, May 2001.

[6] S. Boll, W. Klas, and U. Westermann. Multimedia Document Formats — Sealed Fate or Setting Out for New Shores? *Multimedia - Tools and Applications*, 11(3), 2000.

[7] D. Brickely and R.V. Guha. Resource Description Framework (RDF) Vocabulary Description Language 1.0: RDF Schema. W3C Working Draft, World Wide Web Consortium (W3C), April 2002.

[8] H. Chang, T. Hou, A. Hsu, and S. Chang. Tele-Action Objects for an Active Multimedia System. In *Proc. of the International Conference on Multmedia Computing and Systems (ICMCS 1995)*, Ottawa, Canada, 1995.

[9] S. Chang and T. Znati. Adlet: An Active Document Abstraction for Multimedia Information Fusion. *IEEE Transactions on Knowledge and Data Engineering*, 13(1), 2001.

[10] R. Daniel, C. Lagoze, and S. Payette. A Metadata Architecture for Digital Libraries. In *Proc. of the Advances in Digital Libraries Conference*, Santa Barbara, California, 1998.

[11] D. Fensel. *Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce*. Springer, Heidelberg, 2001.

[12] J. Ferraiolo, F. Jun, and D. Jackson. Scalable Vector Graphics (SVG) 1.1. W3C Recommendation, World Wide Web Consortium (W3C), January 2003.

[13] ISO/IEC IS 13522-5. Information Technology – Coding of Hypermedia Information - Part 5: support for Base-Level Interactive Applicatons. ISO/IEC International Standard, International Organization for Standardization/International Electrotechnical Commission (ISO/IEC), 1996.

[14] ISO/IEC JTC 1/SC 29/WG 11. Information Technology – Multimedia Content Description Interface – Part 5: Multimedia Description Schemes. ISO/IEC Final Draft International Standard 15938-5:2001, International Organization for Standardization/International Electrotechnical Commission (ISO/IEC), October 2001.

[15] ISO/IEC JTC 1/SC 34/WG 3. Information Technology – Hypermedia/Time-Based Structuring Language (HyTime). ISO/IEC International Standard 15938-5:2001, International Organization for Standardization/International Electrotechnical Commission (ISO/IEC), September 1997.

[16] ISO/IEC JTC 1/SC 34/WG 3. Information Technology – SGML Applications – Topic Maps. ISO/IEC International Standard 13250:2000, International Organization for Standardization/International Electrotechnical Commission (ISO/IEC), February 2000.

[17] ISO/JTC 1/SC 32/WG 2. Conceptual Graphs. ISO/IEC International Standard, International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) , April 2001.

[18] C. Lagoze, C. Lynch, and R. Daniel. The Warwick Framework: A Container Architecture for Aggregating Sets of Metadata. Technical Report TR 96-1593, Cornell University, Ithaca, New York, 1996.

[19] O. Lassila and R.R. Swick. Resource Description Framework (RDF) Model and Syntax Specification. W3C Recommendation, World Wide Web Consortium (W3C), February 1999.

[20] P.J Leach. UUIDs and GUIDs. Network Working Group Internet-Draft, The Internet Engineering Task Force (IETF), February 1998.

[21] V. Matena and M. Hapner. Enterprise Java Beans^{TM}. Specification version 1.0, Sun Microsystems Inc., March 1998.

[22] D. Newman, A. Patterson, and P. Schmitz. XHTML+SMIL Profile. W3C Note, World Wide Web Consortium (W3C), January 2002.

[23] F. Pereira and T. Ebrahimi, editors. *The MPEG-4 Book*. Pearson Education, California, 2002.

[24] D. Raggett, A. Le Hors, and I. Jacobs. HTML 4.01 Specification. W3C Recommenation, World Wide Web Consortium (W3C), December 1999.

[25] S. Reich, W. Behrendt, C. Eichinger, and M. Schaller. Document Models for Navigating Digital Libraries. In *Proc. of the Kyoto International Conference on Digital Libraries*, Kyoto, Japan, 2000.