Integration of ontological knowledge within the authoring and retrieval of multimedia metaobjects

Sonja Zillner*

Department of Distributed and Multimedia Systems
University of Vienna
Liebiggasse 4/3–4, A-1010 Vienna, Austria
E-mail: sonja.zillner@univie.ac.at
*Corresponding author

Werner Winiwarter

Department of Scientific Computing
University of Vienna
Universitätsstraße 5/9, A-1010 Vienna, Austria
E-mail: werner.winiwarter@univie.ac.at

Abstract: To enable efficient authoring, management and access to multimedia content, media data has to be augmented by semantic metadata and functionality. Semantic representation has to be integrated with domain ontologies to fully exploit domain-specific knowledge. This knowledge can be used within the authoring process and for the efficient management of multimedia content. Also, this knowledge can be used for refining ambiguous user queries by closing the conceptual gap between the user and the information to be retrieved. In our previous research, we have introduced Enhanced Multimedia Metaobjects (EMMOs) as a new approach for semantic multimedia meta modelling, as well as the query algebra EMMA, which is adequate and complete with regard to the EMMO model. This paper illustrates how ontological knowledge can be used within the authoring process of EMMOs, integrated into the EMMO knowledge structures and exploited for refining EMMA queries.

Keywords: web semantics; metadata; ontology; semantic modelling of multimedia content; query algebra; query refinement.

Reference to this paper should be made as follows: Zillner, S. and Winiwarter, W. (2005) ‘Integration of ontological knowledge within the authoring and retrieval of multimedia metaobjects’, *Int. J. Web and Grid Services*, Vol. 1, Nos. 3/4, pp.397–415.

Biographical notes: Sonja Zillner studied mathematics at the University Freiburg in Germany and received her diploma degree in 1999. Since 2000 she has been a member of the scientific staff of the Institute of Distributed and Multimedia Systems, University of Vienna. Her research interests lie in the areas of semantic multimedia content modelling and ontologies.

Professor Dr. Werner Winiwarter holds a tenured position at the Institute of Scientific Computing, University of Vienna. He received an MS degree in 1990, an MA degree in 1992, and a PhD in 1995, all from the University of Vienna. The main research interest of Winiwarter is human language technology. In addition, he also works on data mining, and machine learning, semantic web, information retrieval and filtering, electronic business, digital libraries, and education systems.

1 Introduction

Although more and more multimedia content becomes available, efficient management of multimedia content is still an open and critical issue. Efficient management, in this context, pertains to: the authoring of; comfortable access to; and searching for multimedia resources. To prevent the generation and storage of useless material, it should be assured during the *authoring* of multimedia content that the interrelations between multimedia resources are compatible with commonly accepted knowledge. In order to facilitate the efficient *access* to multimedia content within a distributed environment, the representation of multimedia content should be flexible in a way that – provided that the semantics of a document is preserved – the user can choose between a minimal, compact version being efficient for sending and a maximal version providing optimal browsing and query performance. Text-based keyword search techniques alone are not sufficient for *retrieving* multimedia content efficiently; semantic meta-information has to be integrated within the retrieval process. One way to improve the authoring of, the efficient access to and the search for multimedia content is to enhance multimedia resources by *semantic metamodels* and to integrate them with *domain ontologies.*

To answer these challenges, we have developed Enhanced Multimedia Meta Objects (EMMOs) (Schellner *et al.*, 2003), a novel approach for semantic multimedia content modelling. EMMOs were created within the EU-project CULTOS to model InterTextualThreads (ITTs), which are complex knowledge structures used by the researchers in intertextual studies to share and communicate their knowledge about relationships between cultural artifacts. An EMMO establishes a self-contained piece of multimedia content that indivisibly combines three of the content's aspects. The *media aspect* describes that an EMMO assembles the media objects of which the multimedia content consists of. The *semantic aspect* specifies that an EMMO further encapsulates associations between its media objects and the *functional aspect* defines operations on the content and on the semantic description of an EMMO which can be invoked by applications. Moreover, EMMOs are *tradeable*, they can be packaged and exchanged in their entirety including media, semantics and functionality. Finally, EMMOs constitute *versionable* units that can be concurrently modified and authored within a distributed collaborative environment.

For the efficient retrieval of and access to the multimedia content within EMMOs, we have developed the query algebra *EMMA*, which is adequate and complete with regards to the EMMO model – it provides the required operators to enable the access to the complete information stored within the EMMO model. By defining simple and orthogonal operators that can be combined to build complex queries, EMMA enables efficient query optimisation.

The EMMO model and the EMMA algebra both provide a basis for the integration of a domain ontology. This is because an EMMO establishes a graph-like *knowledge structure* with associations and nodes being labelled by concepts of the domain ontology. Also, EMMA defines *navigational operators* to provide the means to traverse the ontology-labelled associations within an EMMO's graph structure.

The main contribution of this paper is to show how ontological knowledge can be used for the following:

- checking of integrity constraint within the design and authoring process of EMMOs
- enhancing the semantic expressiveness of the knowledge structures defined by EMMOs
- refining EMMA query expressions.

The remainder of the paper is organised as follows. In Section 2, we discuss related approaches and standards. Section 3 gives a brief overview of the EMMO model and its query algebra EMMA. In Section 4, we describe how the ontology knowledge can be integrated into the EMMO model and EMMA algebra. Section 5 discusses different ways of representing ontology knowledge. Section 6 concludes this paper with an outlook on future work.

This paper builds on, revises and extends previous research work published in Zillner and Winiwarter (2004a–b).

2 Related work

By incorporating the media, the semantic and the functional aspects, as well as versioning support in a homogeneous way, the EMMO model constitutes a unique approach to multimedia meta modelling. None of the standards for *multimedia document models*, such as SMIL (Ayars *et al.*, 2001) or SVG (Ferraiolo *et al.*, 2003), and none of the standards for *semantic media description*, such as RDF (Lassila and Swick, 1999) or Topic Maps (ISO/IEC JTC 1/SC 34/WG 3, 2000), addresses all these aspects. Thus, none of the query languages for those standards can fulfill all the requirements regarding the expressiveness of a query language for EMMOs. However, valuable aspects of the design of these query languages have been incorporated into the design of the query algebra EMMA.

Besides the integration of ontological knowledge within the EMMO model, we focus on the enhancement of the expressive power of EMMA's navigational operators by integrating ontology knowledge. Therefore, we analysed query languages for standards for semantic media description, such as RDF or Topic Maps, as to whether they allow for navigation of the graph structure and whether the navigational access can be refined by integrating ontology knowledge.

Existing query languages for RDF, such as RAL (Frasincar *et al.*, 2002) or RQL (Karvounarakis *et al.*, 2002), provide the means to navigate the RDF graph structure and enable the integration of a very simple ontology structure described by RDF Schema (Brickely and Guha, 2002). However, these query languages cannot deal with more elaborate ontology constructs, such as the transitivity or symmetry of relationships. The situation for query languages for *Topic Maps* is quite similar to that of RDF. Approaches such as Tolog (Garshol, 2003), TMPath (Bogachev, 2004), or XTMPath (Barta and Gylta, 2002) provide the means for graph navigation. Only Tolog, however, addresses ontology integration by offering some very limited features, such as querying concept-subconcept relationships.

Although there exist quite a few approaches for ontology-based browsing of text collections, such as QuizRDF (Davies *et al.*, 2004), Spectacle (Flut *et al.*, 2003) or SESQ (Zhang *et al.*, 2003), none of those approaches is suitable for the authoring, the efficient access to and the retrieval of multimedia data. They also do not allow user queries to be refined by drawing inferences from ontological knowledge. However, these approaches offer useful insights into the use of ontology knowledge for visualisation and navigation in browsing interfaces. We will bear these in mind for future work on designing the user interface of an ontology-based search engine.

3 The EMMO model and the EMMA algebra

As mentioned before, an EMMO is a self-contained unit of multimedia content that unifies three aspects – the media, the semantic and the functional aspect. The EMMO also provides versioning support. Figure 1 shows the EMMO 'Dracula', which we will use as a running example throughout this paper. The formal bases of the EMMO model are *entities*, which occur in four different specialisations:

1. *ontology objects* represent concepts of an ontology
2. *logical media parts* represent media objects or parts of media objects, such as video scenes or book chapters
3. *associations* model binary relationships
4. *EMMOs* aggregate semantically related entities.

Each entity is globally and uniquely identified by its OID and described by an arbitrary number of attribute-value pairs. The attribute name is a concept of the ontology. For example, the ontology object *Director* labels an attribute of the logical media part 'Dracula', which has the string value 'F. Coppola'. For enabling *versioning support*, each entity can refer to an arbitrary number of preceding and succeeding versions.

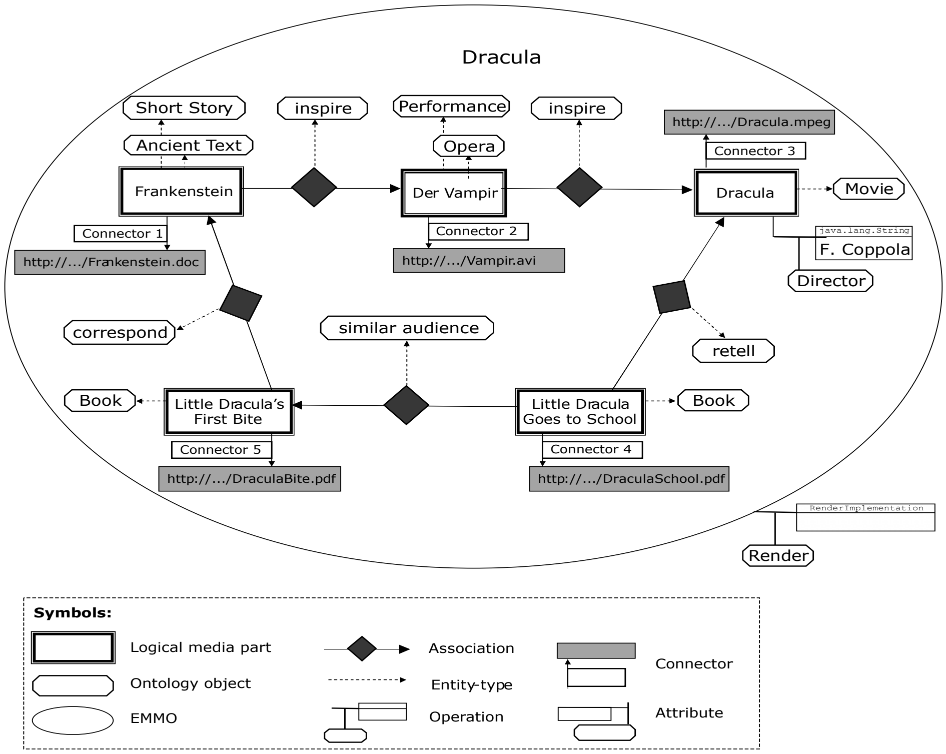
To model an EMMO's *media aspect*, each logical media part is characterised by a set of connectors with each connector pointing to a physical representation of the media object. For example, the EMMO 'Dracula' contains the text documents 'Frankenstein.doc', 'DraculaSchool.pdf', and 'DraculaBite.pdf', as well as the two videos 'Vampir.avi' and 'Dracula.mpeg'.

By specifying a set of operations, EMMOs address the *functional aspect*, *e.g.*, EMMO 'Dracula' provides a rendering operation, which returns a presentation of the EMMO's content in different formats, such as HTML or PDF.

For the representation of an EMMO's *semantic aspect*, each EMMO is described by a *nodes* set. This set specifies all the entities contained within an EMMO. Semantic relationships between entities are modelled as directed associations by indicating a source and a target entity for which the relationship holds. Therefore, EMMOs constitute a graph-like knowledge structure with links and edges being labelled by ontology objects representing concepts of the domain ontology. The media objects contained within

EMMO 'Dracula' are digital manifestations of the ancient text 'Frankenstein', the German opera 'Der Vampir', the movie 'Dracula' directed by Francis Ford Coppola and two children's books 'Little Dracula Goes to School' and 'Little Dracula's First Bite'. The types of the media objects are defined through a reference to concepts of the domain ontology, such as *Ancient Text* or *Performance.* By labelling the associations with relational concepts from the ontology, we can indicate that the historical text 'Frankenstein' *inspired* the German opera 'Der Vampir', which again *inspired* the movie 'Dracula'. Furthermore, we can express that both books 'Little Dracula Goes to School' and 'Little Dracula's First Bite' address a *similar audience*, 'Little Dracula Goes to School' *retells* the 'Dracula' movie, and 'Little Dracula's First Bite' *corresponds* to the 'Frankenstein' text.

Figure 1 EMMO 'Dracula' $(e_{dracula})$



Finally, EMMOs can be serialised into a bundle that completely encompasses all three aspects and versioning information. This means that EMMOs are transferrable in their entirety between different EMMO providers. Thus, EMMOs provide the basis for the *distributed*, *collaborative authoring* of multimedia enhanced knowledge structures.

To enable the efficient retrieval of EMMOs, we developed the *query algebra EMMA*, which is adequate and complete with regards to the EMMO model. By providing simple and orthogonal operators, which can be combined to formulate more complex queries, EMMA enables efficient query optimisation. EMMA defines five general classes of query operators:

1 *Extraction operators* provide the means to access all the attributes of the entities within the EMMO model.

2 *Selection predicates* allow the selection of only those entities satisfying a specific characteristic.

3 *Construction operators* enable the modification, combination and creation of new EMMOs.

4 *Join operator* relates several entities or EMMOs with a join condition.

5 *Navigational operators* enable the navigation along an EMMO's semantic graph structure.

Therefore, the navigational operators provide the basis for ontology-based query refinement.

Both the EMMO model and the EMMA algebra provide a sound basis for the integration of ontological knowledge. This will be explained in more detail in Section 4.

4 Integration of ontological knowledge

By providing a shared and common understanding of a domain that can be communicated between people and application systems, ontologies facilitate the sharing and reuse of knowledge (Fensel, 2001). Ontologies describe concepts, relationships, and constraints in the domain of discourse. The integration of ontology knowledge into the EMMO model and the EMMA algebra has three appealing benefits:

1 Ontological knowledge can be used for checking integrity constraints during the design and *authoring process of EMMOs*. For example, it can be used to store only associations in the database which conform to the specified types for source and target entities.

2 Ontological knowledge can be incorporated within the *EMMO model* by extending the graph structure of an EMMO with additional associations. For example, if two concepts are stated to be inverse to each other, such as *retell* and *is-retold*, then for each association of one of the two types, an association classified by its inverse counterpart can be added.

3 Knowledge inherent in a domain ontology can be seamlessly integrated into *EMMA queries.* Therefore, the user can pose imprecise queries, which are refined by drawing inferences over the ontological knowledge. For example, if the user asks for all media objects which had been *inspired* by the ancient text 'Frankenstein', the user should also receive media objects which were indirectly *inspired* by the ancient text, such as the movie 'Dracula'. This can be accomplished if the transitivity of the ontology object *inspire* is known, that is, it is defined in the ontology.

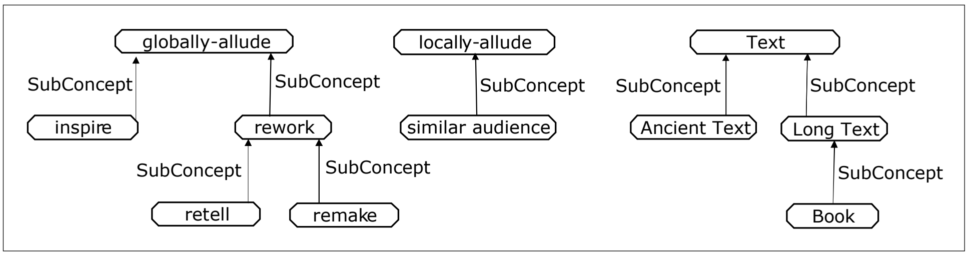
In the following section, we define an ontology structure suitable for the EMMO model, describe how the most common modelling constructs used in standard ontology languages, such as DAML+OIL (Connolly *et al.*, 2001) or OWL (Schneider *et al.*, 2004), can be represented within this structure and exemplify how the ontology knowledge can be integrated into the authoring process of EMMOs (Section 4.1), into the knowledge structures described by the EMMO model (Section 4.2) and into the EMMA queries (Section 4.3).

The definition of an ontology structure for EMMOs was inspired by the ontology structure definition in Maedche (2002). Any concept of the ontology which is used for labelling entities in the EMMO model is represented as an ontology object within the EMMO model. As the EMMO model treats associations as first-class objects, ontology objects can be used for labelling both the nodes and the edges in an EMMO's graph structure. We specify an *ontology structure* suitable for the EMMO model as three-tuple $O = \{\Theta, H^{\Theta}, A^{O}\}$ consisting of the following:

- A *set of ontology objects* Θ representing the concepts of the ontology.
- A *concept hierarchy* H^{Θ} describing the subclass relationship between ontology objects, *i.e.*, H^{Θ} *is* a directed relation $H^{\Theta} \subseteq \Theta \times \Theta$ with $H^{\Theta}(o_1, o_2)$ expressing that o_1 is a subconcept of o_2.
- A set of *ontology axioms* A^{O}, expressed in first-order logic.

Figure 2 illustrates a small portion of the Ontology of Intertextuality used in the CULTOS project as defined in Benari *et al.* (2003).

Figure 2 Extract from the ontology of intertextuality



An ontology suitable for the integration into the EMMO model and EMMA algebra has to distinguish between *object concepts* and *relational concepts.* Object concepts are used for labelling the nodes. Relational concepts are used for labelling the associations within an EMMO's graph structure. For example, the Ontology of Intertextuality defines object concepts for describing media objects, such as the concepts *Text* or *Performance*, and relational concepts for describing relationships holding between media objects, such as the relational concept *inspire* can be used for describing the fact that an ancient source text *inspires* a particular performance.

The set of ontology axioms A^{O} allows to specify properties and restrictions of concepts. It also allows to define properties of relationships between concepts. Thus, we can specify that some specific ontology objects are dedicated to describe associations within the EMMO model, that is, it represents relational concepts. For example:

$$(\{o_{globally-allude}, o_{inspire}, o_{rework}, o_{retell}, o_{remake}, o_{locally\text{-}allude}, o_{similar}\} \subseteq PX) \in A^{O}, \quad (1)$$

with $PX = \{o \in \Theta | \forall w \in \Omega \wedge o \in types(w) \rightarrow w \in \Lambda\}$ describing the set *relational concepts*, Ω the set of all entities, Λ the set of all associations and *types(w)* the set of ontology objects labelling the entity *w*. In a similar way, we can specify that some ontology objects are used exclusively for describing nodes of the EMMO graph structure. For example:

$$(\{o_{Text}, o_{AncientText}, o_{LongText}, o_{Book}\} \subseteq OX) \in A^O, \tag{2}$$

with $OX = \{o \in \Theta | \forall w \in \Omega \wedge o \in types(w) \rightarrow w \notin \Lambda\}$ describing the set of *object concepts.* Furthermore, within the set of ontology axioms, we can define the transitivity of the concept hierarchy, that is:

$$(\forall o_1, o_2, o_3, \in \Theta\ H^{\Theta}(o_1,o_2) \wedge H^{\Theta}(o_2,o_3) \rightarrow H^{\Theta}(o_1,o_3)) \in A^O. \tag{3}$$

Based on this axiom, we can now infer from the ontology that the concept *Book*, which is a direct subconcept of *Long Text*, is also a subconcept of *Text*, which is the superconcept of *Long Text.*

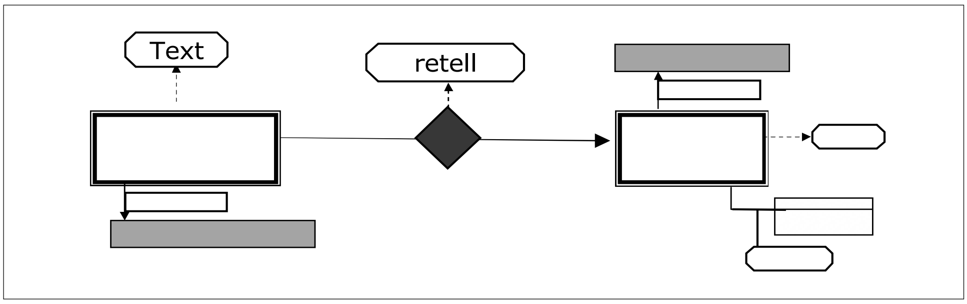
4.1 Integration of ontological knowledge within the authoring of EMMOs

During the design and authoring process of EMMOs, we can use ontological knowledge for checking integrity constraints. This means that only associations that coincide with the specified types regarding source and target entity can be stored in the database. For instance, within an ontology structure one can specify that associations of type *retell* describe binary relationships pointing from entities of type *Text* to entities of arbitrary type. That is:

$$(Domain(o_{retell}) = \{o_{text}\}) \in A^O, \tag{4}$$

with $Domain(o) = \{x \in \Theta | \forall a \in \Lambda \wedge o \in types(a) \rightarrow x \in types(source(a))\}$ describing the set of *domain concepts*, which are the concepts used for classifying the source entity of associations of type *o* and *source(a)* denoting the source entity of association *a* (see Figure 3).

Figure 3 Association 'retell' with source entity of type 'Text'



Let us assume that a user intends to store the EMMO 'Dracula' (see Figure 1) in the database. Due to integrity constraints checking based on Axiom 4, the EMMO's *retell* association, which in this example specifies a source entity of type *Book* but not of type *Text*, will be removed from the EMMO's nodes before storing the EMMO.

In order to reflect the ontological knowledge about the concept hierarchy during the checking of integrity constraints, such as the knowledge that *Book* is a subconcept of *Long Text* which again is a subconcept of *Text*, we can specify the following axiom:

$$(\forall\, o \in PX\, \forall\, O \subseteq \Theta\; Domain(o) = O \Rightarrow Domain(o) = SubConcepts(O)) \;\in\; A^{O}, \quad (5)$$

with $SubConcepts(O) = \{x \in \Theta \mid \exists o \in O(x,o) \in H^{\Theta}\}$ describing the set of all subconcepts of a set of ontology objects *O*.

By integrating Axioms 3–5 within the authoring process of EMMOs, EMMO ‘Dracula’ can now be stored in the database without removing the *retell* association (see Figure 4).

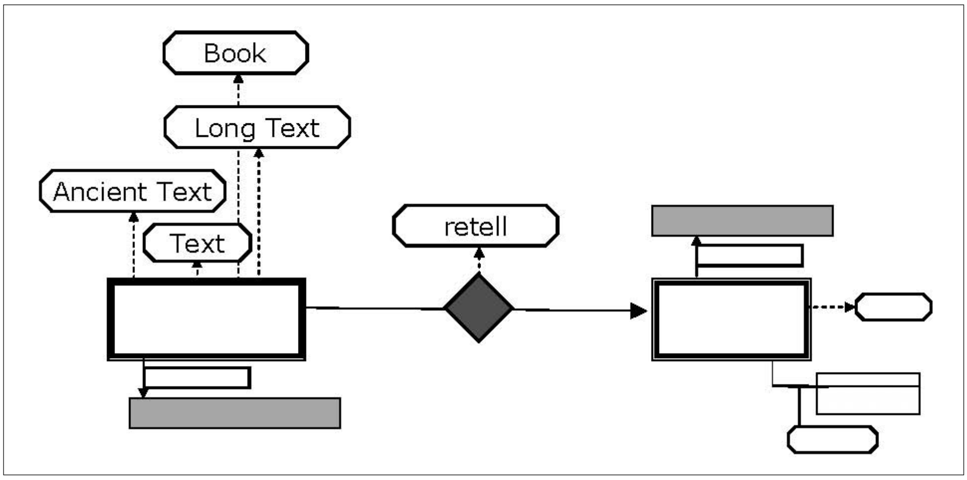
In a similar way, we define axioms for constraints on *range concepts*, which are concepts used for classifying the target entity of associations. For example:

$$(Range(o_{retell}) = \{o_{movie}\}) \in A^{O} \text{ and} \quad (6)$$

$$(\forall\, o \in PX\; \forall\; O \subseteq \Theta\; Range(o) = O \Rightarrow Range(o) = SubConcepts(O)) \in A^{O}, \quad (7)$$

with $Range(o) = \{x \in \Theta \mid \forall\, a \in \Lambda \wedge o \in types(a) \rightarrow x \in types(target(a))\}$ describing the set of concepts which can be used for classifying the target entity of associations of type *o* and *target(a)* denoting the target entity of association *a.*

Figure 4 Association ‘retell’ with source entity of type $SubConcepts(\{o_{text}\})$



4.2 Integration of ontological knowledge into the EMMO model

Ontological knowledge can be integrated into the EMMO model by extending the graph structure with additional associations. In the following, we will exemplify how the ontological knowledge about the subconcept hierarchy, as well as about inverse, transitive and symmetric relational concepts, can be integrated within the EMMO model. Furthermore, we will introduce the axioms describing this ontological knowledge. All the examples are based on the ontology structure illustrated in Figure 2 and the EMMO ‘Dracula’ depicted in Figure 1.

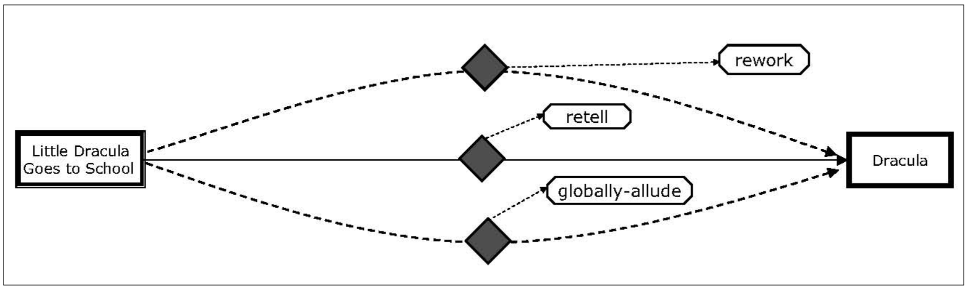
4.2.1 Integrating the knowledge about the subconcept hierarchy

In Axiom 3, we define the transitivity of the concept hierarchy. By adding the axiom:

$$(\{(o_{rework}, o_{globally\text{ - }allude}), (o_{retell}, o_{rework}), (o_{remake}, o_{rework}),\ldots\} \subseteq H^{\Theta}) \in A^{O}, \qquad (8)$$

we specify, that the concepts *remake* and *retell* are subconcepts of *rework*, which is again a subconcept of *globally-allude*. The explicit integration of this knowledge into the EMMO model can be realised by adding the associations $(l_{school} \xrightarrow{o_{rework}} l_{dracula})$ and $(l_{school} \xrightarrow{o_{globally-allude}} l_{dracula})$ to the EMMO 'Dracula' (see Figure 5).

Figure 5 Integrating the knowledge about *retell's* superconcepts



4.2.2 Integrating the knowledge about transitive concepts

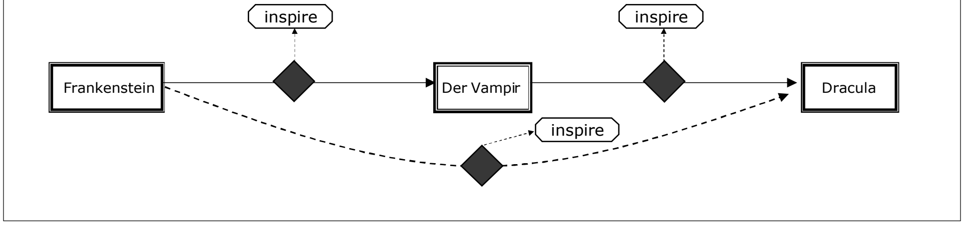
Within the ontology axioms, we can also define *transitive concepts*. These are relational concepts for which an iteration of the navigation along the corresponding associations can be defined without changing the semantics of the concept. For example:

$$(o_{inspire} \in \Theta_{TRANS}) \in A^{O}, \qquad (9)$$

with $\Theta_{TRANS} = \{o \in PX| \; \forall \; a_1, \quad a_2 \in I(o) \;\; target(a_1) = source(a_2) \rightarrow \exists a_3 \in I(o)$ $(source(a_3) = source(a_1) \wedge target(a_3) = target(a_2))\}$ describing the set of all transitive ontology objects, $I(o) = \{w \in \Omega \,|o \in types(w)\}$ the set of all entities labelled by the ontology object o and $source(a)$ and $target\ (a)$ the source and target entities of association a.

To integrate the knowledge that the concept *inspire* is a transitive concept into the EMMO model, we add the association $(l_{frank} \xrightarrow{o_{inspire}} l_{dracula})$ to the EMMO 'Dracula' (see Figure 6).

Figure 6 Integrating the knowledge that *inspire* is a transitive concept



4.2.3 Integrating the knowledge about symmetric concepts

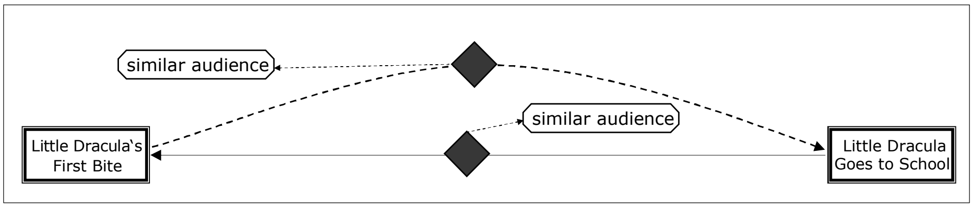
In a similar way, we express *symmetric concepts*. These are relational concepts for which all associations can be traversed in both directions. This means that source and target entities can be exchanged without changing the semantics of the concept. For example:

$$(o_{similar} \in \Theta_{\mathrm{SYM}}) \in \mathrm{A}^{\mathrm{O}}, \quad (10)$$

with $\Theta_{\mathrm{SYM}} = \{o \in \mathrm{PX} | \ \forall \ a_1 \in I(o) \ \exists a_2 \in I(\mathrm{o})(source(a_1) = target(a_2) \wedge source(a_2) = target(a_1))\}$ describing the set of all symmetric ontology objects.

To explicitly incorporate the knowledge about *similar audience* being a symmetric concept, we add the association $(l_{bite} \xrightarrow{o_{similar}} l_{school})$ to the EMMO 'Dracula' (see Figure 7).

Figure 7 Integrating the knowledge that *similar audience* is a symmetric concept



4.2.4 Integrating the knowledge about inverse concepts

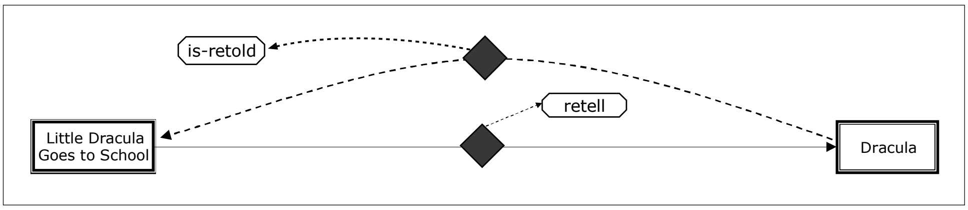
Finally, we can also express that two relational concepts are *inverse* to each other. This means that if an association is labelled with the inverse concept, then source and target entities have to be exchanged to keep the semantics intact. For example:

$$((o_{retell}, o_{is-retold}) \in \Theta_{\mathrm{INV}}) \in \mathrm{A}^{\mathrm{O}}, \quad (11)$$

with $\Theta_{\mathrm{INV}} = \{(o_1, o_2) \in \mathrm{PX} \times \mathrm{PX} | \ \forall \ a_1 \in I(o_1) \ \exists a_2 \in I(o_2)(source(a_1) = target(a_2) \wedge source(a_2) = target(a_1))\}$ describing the set of all pairs of inverse ontology objects.

The fact that the concepts *retell* and *is-retold* are two inverse concepts can be expressed by adding the association $(l_{dracula} \xrightarrow{o_{is-retold}} l_{school})$ to the EMMO 'Dracula' (see Figure 8).

Figure 8 Integrating the knowledge that *retell* and *is-retold* are inverse concepts



To improve the access to EMMOs by integrating ontological knowledge into the EMMO model, an EMMO's graph structure has to be extended. This can be done by additional associations according to the ontological axioms. Therefore, the EMMO is *inflated.*

On the other hand, for enabling efficient authoring and exchanging of EMMOs within a distributed environment, EMMOs are required to be of compact size. To realise this, similar to the inflation of EMMOs, ontological knowledge can be used for *deflating* EMMOs. This means that any redundant association within an EMMO, which can be inferred from the ontology, is removed from an EMMO's nodes.

Thus, there are two different ways of representing EMMOs – a compact *minimum version* being optimal for exchanging EMMOs and a *maximum version* to provide efficient access to EMMOs. The decision as to which of the two versions is preferable for a given situation depends on several factors, such as the requirements of the application scenario, the depth and size of the ontology and the size of an EMMO. Typical decision criteria are response time constraints for retrieval, storage limitations in mobile devices and restricted bandwidth for network transmissions.

This leads to the problem on how to exploit ontological knowledge for the retrieval of multimedia content in situations that would demand the use of an EMMO's minimal version. To solve this dilemma, we have extended the query algebra EMMA. We provide means for integrating ontological knowledge within the processing of EMMA's navigational queries by adding alternative navigation paths (see Section 4.3). In this way, instead of *explicitly extending* an EMMO's graph structure, the graph structure now is *implicitly extended* during query execution.

4.3 Integration of ontological knowledge within EMMA query processing

EMMA's navigational operators enable the navigation along an EMMO's semantic graph structure. Thus, they provide the basis for ontology-based query refinement. In the following, we will first provide a detailed description of the navigational operators and then illustrate how EMMA query processing can be enhanced by integrating ontological knowledge.

Navigation through an EMMO's graph structure is controlled by a *navigation path* defined as a set of *sequences* of ontology objects. A mapping for each ontology object in a sequence to the corresponding association within an EMMO defines the traversal of the graph structure.

We have already defined *regular path expressions* over ontology objects for describing the syntax of a navigation path. Navigational operators take a regular path expression as input and specify how those syntactic expressions are applied to navigate the graph structure. For a given EMMO, start entity, and regular path expression for example, the navigational operator *JumpRight* returns the set of all entities that can be reached by traversing the navigation path in the right direction. This happens by following associations from source to target entities. Applying the operator *JumpRight* to the EMMO 'Dracula' ($e_{dracula}$) in Figure 1, the starting entity 'Frankenstein' (l_{frank}) and the primitive regular path expression consisting of one single ontology object *inspire* ($o_{inspire}$) yields the logical media part representing the video of the opera 'Der Vampir' (l_{vampir}):

$$JumpRight(e_{dracula}, l_{frank}, o_{inspire}) = \{l_{vampir}\}.$$

In addition to *one single ontology object*, there exist two other primitive regular path expressions:

1 The empty expression 'ε' refers to the empty entity. It is interpreted by a navigational operator as absence of movement.

2 The *wildcard expression* '-' refers to any arbitrary ontology object. It is interpreted by a navigational operator as following any arbitrary association regardless of the labelling.

There exist two ways of combining regular path expressions:

1 Regular path expressions can be *concatenated* to create a longer navigation path.

2 The *union operator* '|' allows to treat several regular path expressions as alternative branches.

Finally, we have defined four unary operators for the modification of regular path expressions:

1 Adding the operator '?' to a regular path expression specifies its *optionality.*

2 Adding the operator '+' to a regular path expression specifies an *iteration* of path expressions, which is interpreted as navigation along the same regular path expression any number of times but at least once.

3 Adding the *Kleene Star operator* '*' to a regular path expression specifies an iteration of path expressions. It is interpreted as navigation along the same regular path expression any number of times (including the absence of movement).

4 Adding the operator '–' to a regular path expression expresses the *inversion* of the regular path expression, which is the change of direction of navigation.

Table 1 shows examples of applying the *JumpRight* operator with different types of regular path expressions to the EMMO 'Dracula'($e_{dracula}$). The symbols l_{frank}, l_{vampir}, $l_{dracula}$, l_{bite} and l_{school} designate the logical media parts 'Frankenstein', 'Der Vampir', 'Dracula', 'Little Dracula's First Bite' and 'Little Dracula Goes to School'; $o_{inspire}$, $o_{similar}$, $o_{correspond}$, and o_{retell} the ontology objects *inspire*, *similar audience*, *correspond* and *retell.*

The inversion of a regular path expression, which is the traversal along the opposite direction, can also be expressed with the navigational operator *JumpLeft*. For example:

$$JumpLeft(e_{dracula}, l_{frank}, o_{correspond}) = JumpRight(e_{popular}, l_{frank}, o_{correspond}–) = \{l_{bite}\}.$$

The integration of ontology knowledge into EMMA query processing allows to refine user queries. Thus, a user can pose imprecise queries, which are then refined by drawing inferences over ontological knowledge. Instead of extending the EMMO graph structure (see Section 4.2), we extend the navigational operators in EMMA by adding alternative navigation paths. In the following, we will illustrate how the knowledge captured by the ontology structure illustrated in Figure 2 and described in more detail in Axioms 8–11 can be used for ontology-based query refinement.

Table 1 Examples of applying regular path expressions

Basic pattern	*Example query*	*Query result*
o_i	$JumpRight(e_{dracula}, l_{frank}, o_{inspire})$	$\{l_{vampir}\}$
ε	$JumpRight(e_{dracula}, l_{frank}, \varepsilon)$	$\{l_{frank}\}$
-	$JumpRight(e_{dracula}, l_{school}, -)$	$\{l_{bite}, l_{dracula}\}$
$o_i o_j$	$JumpRight(e_{dracula}, l_{school}, o_{similar}\, o_{correspond})$	$\{l_{frank}\}$
$o_i \vert o_j$	$JumpRight(e_{dracula}, l_{school}, o_{similar} \vert o_{retell})$	$\{l_{bite}, l_{dracula}\}$
$o_i?$	$JumpRight(e_{dracula}, l_{frank}, o_{similar}?o_{inspire})$	$\{l_{vampir}\}$
o_i+	$JumpRight(e_{dracula}, l_{frank}, o_{inspire}+)$	$\{l_{vampir}, l_{dracula}\}$
o_i*	$JumpRight(e_{dracula}, l_{frank}, o_{inspire}*)$	$\{l_{frank}, l_{vampir}, l_{dracula}\}$
o_i-	$JumpRight(e_{dracula}, l_{dracula}, o_{retell}-)$	$\{l_{school}\}$

4.3.1 Integrating the knowledge about the subconcept hierarchy

To include ontological knowledge about the concept hierarchy we add all subconcepts as alternative branches to any ontology object in the regular path expression of a navigational EMMA query. For example, the hierarchical structure defined in Axiom 8 specifies that the concept o_{rework} has two subconcepts o_{remake} and o_{retell}. Therefore, the query:

$$JumpRight(e_{dracula}, l_{school}, o_{rework}) = \varnothing$$

can be expanded to search also for all subconcepts of *rework*, which are *retell* and *remake*:

$$JumpRight(e_{dracula}, l_{school}, o_{rework} | o_{retell} | o_{remake}) = \{l_{dracula}\}.$$

A user requesting all entities which were *reworked* by the book *Little Dracula Goes to School*, now receives a useful answer, which is the logical media part 'Dracula' because the book *retells* the movie.

4.3.2 Integrating the knowledge about transitive concepts

To integrate the knowledge about transitive concepts, we add the operator '+' to any transitive ontology object in the regular path expression of a navigational EMMA query. For example, as the ontology object $o_{inspire}$ is defined as a transitive concept (see Axiom 9), we can expand the EMMA query as follows:

$$JumpRight(e_{dracula}, l_{frank}, o_{inspire}) = \{l_{vampir}\}$$

to the query:

$$JumpRight(e_{dracula}, l_{frank}, o_{inspire}+) = \{l_{vampir}, l_{dracula}\}.$$

Therefore, if a user asks for all entities which were *inspired* by the text 'Frankenstein', the user now receives not only the incomplete result consisting of one logical media part 'Der Vampir', which is directly *inspired* by the text 'Frankenstein', but also the logical media part 'Frankenstein' because it is indirectly *inspired.*

4.3.3 Integrating the knowledge about symmetric concepts

By incorporating the knowledge about symmetric concepts, we extend any symmetric ontology object in the regular path expression of a navigational EMMA query by adding its inversion as an alternative branch. For instance, as the ontology object $o_{similar}$ references a symmetric concept (see Axiom 10), the EMMA query:

$$JumpRight(e_{dracula}, l_{bite}, o_{similar}) = \varnothing$$

is expanded to:

$$JumpRight(e_{dracula}, l_{bite}, o_{similar}) \cup JumpLeft(e_{dracula}, l_{bite}, o_{similar}) = JumpRight(e_{dracula}, l_{bite}, o_{similar}|o_{similar}\text{-}) = \{l_{school}\}.$$

Thus, a user asking for all entities which address a *similar audience* as the book *Little Dracula's First Bite* can now retrieve the information that the book *Little Dracula Goes to School* was written for a *similar audience.* This is so although the corresponding association in the EMMO 'Dracula' points to the opposite direction.

4.3.4 Integrating the knowledge about inverse concepts

The integration of the knowledge that an ontology object has an inverse concept is achieved through the extension of the ontology object in any regular path expression of a navigational EMMA query. This is done by adding the inversion of the inverse concept as alternative branch. For example, the knowledge about the two concepts o_{retell} and $o_{is-retold}$ being inverse to each other (see Axiom 11) is reflected by expanding the EMMA query as follows:

$$JumpRight(e_{dracula}, l_{dracula}, o_{is-retold}) = \varnothing$$

to the query:

$$JumpRight(e_{dracula}, l_{dracula}, o_{is-retold}) \cup JumpLeft(e_{dracula}, l_{dracula}, o_{retell}) = JumpRight(e_{dracula}, l_{dracula}, o_{is-retold}|o_{retell}-) = \{l_{school}\}.$$

As a result, if a user wants to know about any entity that *is-retold* by the movie 'Dracula', we can now provide the satisfactory answer *Little Dracula Goes to School* because the EMMO 'Dracula' specifies that this book *retells* the movie.

5 Representation of ontology structures

There exist different ways of representing ontology structures, which, although having the same expressiveness, are designed for different purposes. In the following, we will show three different ways of representing the ontology structure as illustrated in Figure 2 and specified in more detail by Axioms 1–11:

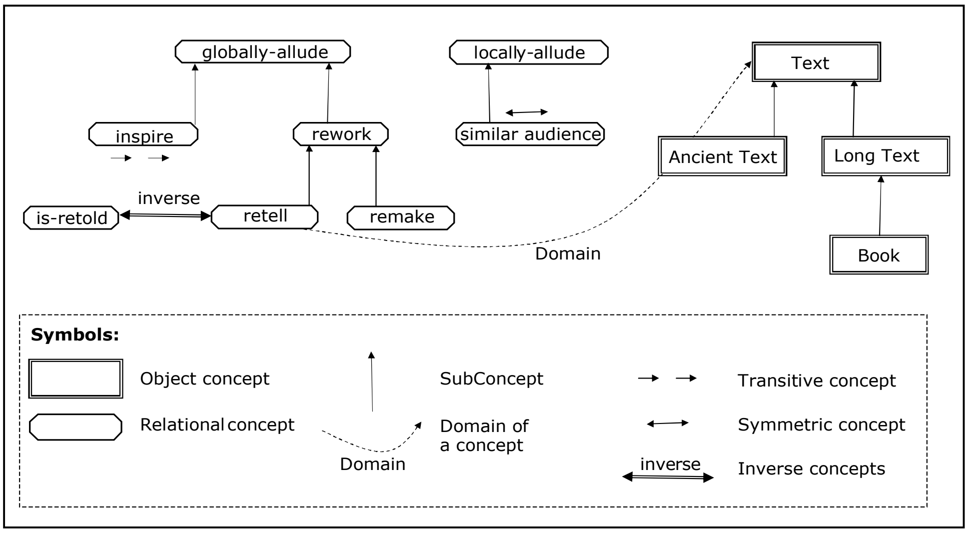
1 The *graphical representation* enhances human readability.

2 The *OWL representation* addresses the standardisation efforts in the context of the semantic web initiative (Berners-Lee *et al.*, 2001).

3 The *EMMO representation* of the ontology structure enables the seamless integration of ontological knowledge into the EMMO model.

Figure 9 shows the graphical representation of the ontology structure in Figure 2. The relational concept *inspire* is marked as transitive. The relational concept *similar audience* is marked as symmetric. And the relational concepts *retell* and *is-retold* are marked as being inverse to each other. Additionally, the object concept *Text* is specified as domain concept for the relational concept *retell.*

Figure 9 Graphical representation of the ontology of intertextuality



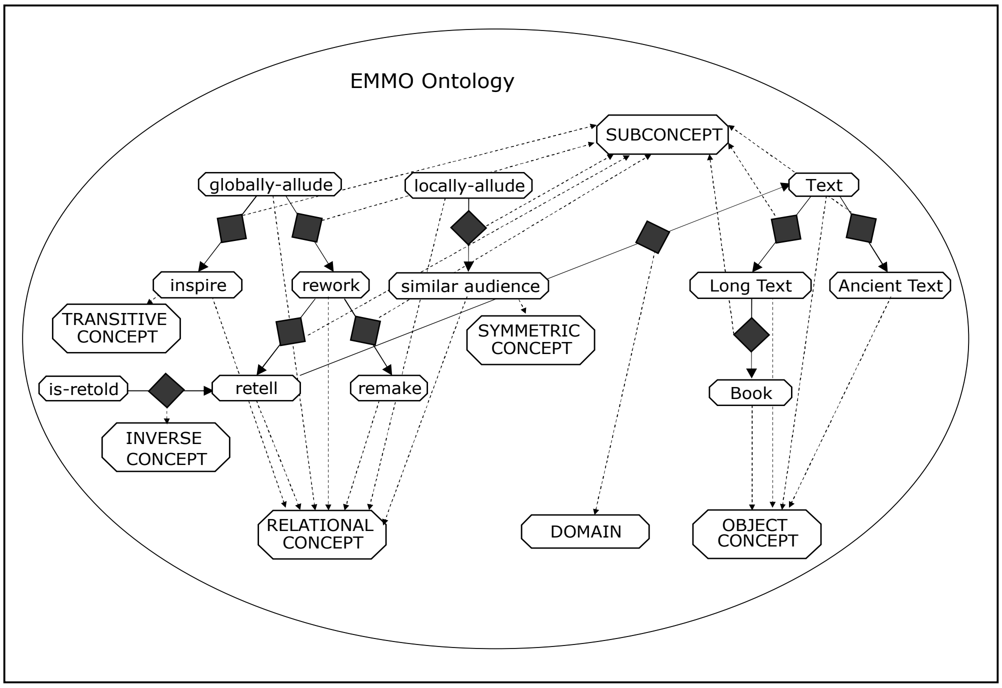
Since DAML+OIL does not provide modelling constructs for symmetric properties, it is not an adequate representation language for ontology structures. Therefore, we used OWL, which specifies all the modelling constructs used within an ontology structure. These are the constructs for expressing transitive, symmetric and inverse concepts. We used Protege-2000 (Stanford Medical Informatics, 2004) as authoring tool for creating the domain ontology and imported the resulting OWL description into the EMMO environment. Figure 10 shows the OWL representation for the ontology in Figure 9.

However, by representing the ontology in a standard format, such as OWL, more complex inferences drawn from the ontology knowledge cannot be integrated into EMMA queries. Therefore, we plan to develop our own ontology description language which is compatible with the EMMO model allowing for sophisticated reasoning on EMMOs. Figure 11 shows the EMMO representation of the ontology structure, which refers to how the ontology structure can be represented within the EMMO model. It is important to mention that the EMMO 'Ontology' uses some 'predefined' *meta ontology objects* corresponding to classical ontology constructs, such as *subconcept*, *relational concept*, *inverse concept* or *domain.* These are again used to classify other ontology objects. For instance, to indicate that the ontology object *inspire* is a transitive, relational concept, it is typed by the meta ontology objects *transitive concept* and *relational concept.* To express that there is a subconcept relationship between the ontology objects *Text* and *Ancient Text*, an association of type *subconcept* between those two ontology objects is established. Finally, to describe that any *retell* association only allows entities of type *Text* as its source entity, an association of type *domain* pointing from the ontology object *retell* to the ontology object *Text* is contained within EMMO 'Ontology'.

Figure 10 OWL representation of the ontology of intertextuality

```
<rdf:RDF
     xmlns:rdf ="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
     xmlns:owl="http://www.w3.org/2002/07/owl#">
    <rdf:Class rdf:ID="Text"/>
    <rdf:Class rdf:ID="AncientText">
        <rdfs:subClassOf rdf:resource="#Text"/></rdf:Class>
    <rdf:Class rdf:ID="LongText">
        <rdfs:subClassOf rdf:resource="#Text"/></rdf:Class>
    <rdf:Class rdf:ID="Book">
        <rdfs:subClassOf rdf:resource="#LongText"/></rdf:Class>
    <rdf:Property rdf:ID="globally-allude"/>
    <rdf:Property rdf:ID="rework">
        <rdfs:subPropertyOf rdf:resource="#globally-allude"/></rdf:Property>
    <owl:TransitiveProperty rdf:ID="inspire">
        <rdfs:subPropertyOf rdf:resource="#globally-allude"/></owl:TransitiveProperty>
    <rdf:Property rdf:ID="remake">
        <rdfs:subPropertyOf rdf:resource="#rework"/> </rdf:Property>
    <rdf:Property rdf:ID="retell">
        <rdfs:subPropertyOf rdf:resource="#rework"/>
        <rdfs:domain rdf:resource="#Text"/></rdf:Property>
    <rdf:Property rdf:ID="is-retold">
        <owl:inverseOf rdf:resource="#retell"/></rdf:Property>
    <rdf:Property rdf:ID="locally-allude"/>
    <owl:SymmetricProperty rdf:ID="similar">
        <rdfs:subPropertyOf rdf:resource="#locally-allude"/></owl:SymmetricProperty>
</rdf:RDF>
```

Figure 11 EMMO representation of the ontology of intertextuality ($e_{ontology}$)



The contribution of an ontology description language is to define the semantics of those and many more meta ontology objects. The representation of the ontology structure within the EMMO model bears two major advantages. First, instead of having to rely on Protege-2000 as an external ontology authoring tool, we can now use the EMMO authoring environment also for the development of the domain ontology. Second, EMMA operators can now be used to draw inferences from the ontological knowledge. Thus, the seamless integration of ontological knowledge within the authoring, processing and querying of EMMOs can be realised.

6 Conclusion

Both EMMOs (a new approach for the semantic modelling of multimedia content) and the query algebra EMMA (enables the access to the knowledge captured by EMMOs) provide a basis for the integration of ontology knowledge into multimedia knowledge management. In this paper, we have illustrated how ontological knowledge can be used for checking integrity constraints within the design and authoring process of EMMOs. We have also illustrated how ontological knowledge can be used to inflate and deflate the knowledge structures described by EMMOs. Finally, we have also illustrated how ontological knowledge can be used for refining EMMA query expressions. We have finished the implementation of the EMMO model and EMMA algebra. We are currently in the process of compiling a comprehensive set of use cases for the performance evaluation of the ontological processing. Our future work will focus on the development of an ontology description language that is fully compatible with the EMMO model. To enable the integration of any arbitrary ontology structure, our intention is the development of an ontology description language with the same expressiveness as the standard ontology description language OWL. We also intend to provide a converter between the two languages. This, on the one hand, guarantees full compatibility with other ontology creation initiatives while, on the other hand, preserves the seamless integration of ontological knowledge into the EMMO authoring environment to unleash its full potential for efficient authoring, management and retrieval of multimedia content.

References

Ayars, J., *et al.* (2001) 'Synchronized Multimedia Integration Language (SMIL2.0)', *W3C Recommendation, World Wide Web Consortium (W3C)*, August.

Barta, R. and Gylta, J. (2002) *XTM::path – Topic Map Management, XPath Like Retrieval and Construction Facility*, Online Article, Available at http://cpan.uwinnipeg.ca/htdocs/XTM/XTM/Path.html

Benari, M., *et al.* (2003) 'Proposal for a standard ontology of intertextuality', *Public Deliverable Version 2.0, CULTOS Consortium and Project Planning*, June.

Berners-Lee, T., Hendler, J. and Lassila, O. (2001) 'The semantic web', *Scientific American*, May.

Bogachev, D. (2004) *TMPath – Revisited*, Online Article, Available at http://www.homepage.mac.com/dmitryv/TopicMaps/TMPath/TMPathRevisited.html

Brickely, D. and Guha, R.V. (2002) 'Resource Description Framework (RDF) vocabulary description language 1.0: RDF schema', *W3C Working Draft, World Wide Web Consortium*, April.

Connolly, D., *et al.* (2001) 'DAML+OIL (March 2001) reference', *W3C Note, World Wide Web Consortium*, December.

Davies, J., Weeks, R. and Krohn, U. (2004) 'QuizRDF: search technology for the semantic web', *Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04)*, Big Island, Hawaii.

Fensel, D. (2001) *Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce*, Springer, Heidelberg.

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

Flut, C., *et al.* (2003) 'Spectacle', *Towards the Semantic Web: Ontology-driven Knowledge Management*, West Sussex, UK.

Frasincar, F., *et al.* (2002) 'RAL: an algebra for querying RDF', *Proceedings of the Third International Conference on Web Information Systems Engineering (WISE 2002)*, Singapore.

Garshol, L. (2003) 'Tolog 0.1', *Ontopia Technical Report*, Ontopia.

ISO/IECJTC1/SC34/WG3 (2000) 'Information technology – SGML applications – topic maps', *ISO/IEC International Standard 13250:2000, International Organization for Standardization/InternationalElectrotechnical Commission (ISO/IEC)*, February.

Karvounarakis, G., *et al.* (2002) 'RQL: a declarative query language for RDF', *Proceedings of the 11th International World Wide Web Conference (WWW 2002)*, Honolulu, Hawaii.

Lassila, O. and Swick, R.R. (1999) 'Resource Description Framework (RDF) model and syntax specification', *W3C Recommendation, World Wide Web Consortium*, February.

Maedche, A. (2002) *Ontology Learning for the Semantic Web*, Kluwer Academic Publishers, Massachusetts, USA.

Schellner, K., Westermann, U., Zillner, S. and Klas, W. (2003) 'CULTOS: towards a world-wide digital collection of exchangeable units of multimedia content for intertextual studies', *Proceedings of the Conference on Distributed Multimedia Systems (DMS 2003)*, Miami, Florida.

Schneider, P., Hayes, P. and Horrocks, I. (2004) 'OWL web ontology language semantics and abstract syntax', *W3C Recommendation, World Wide Web Consortium*, February.

Stanford Medical Informatics (2004) *What is Protege-2000?*, Online documentation, Stanford University School of Medicine, Available at http://www.protege.stanford.edu/

Zhang, Z., *et al.* (2003) 'An ontology-based method for querying the web data', *Proceedings of the 17th International Conference on Advanced Information Networking and Applications (AINA'03)*, Xi'an, China.

Zillner, S. and Winiwarter, W. (2004a) 'Integrating ontology knowledge into a query algebra for multimedia meta objects', *Proceedings of the Fifth International Conference on Web Information Systems Engineering (WISE 2004)*, Brisbane, Australia.

Zillner, S. and Winiwarter, W. (2004b) 'Ontology-based query refinement for multimedia meta objects', *Proceedings of the Sixth International Conference on Information Integration and Web Based Applications and Services (iiWAS 2004)*, Jakarta, Indonesia.