# Enabling Access to Linked Media with SPARQL-MM*

Thomas Kurz
Salzburg Research
Jakob-Haringer-Strasse 5/III
5020 Salzburg, Austria
thomas.kurz@
salzburgresearch.at

Kai Schlegel
University of Passau
Innstrasse 43
94032 Passau
kai.schlegel@
uni-passau.de

Prof. Dr. Harald Kosch
University of Passau
Innstrasse 43
94032 Passau
harald.kosch@
uni-passau.de

## ABSTRACT

The amount of audio, video and image data on the web is immensely growing, which leads to data management problems based on the hidden character of multimedia. Therefore the interlinking of semantic concepts and media data with the aim to bridge the gap between the document web and the Web of Data has become a common practice and is known as Linked Media. However, the value of connecting media to its semantic meta data is limited due to lacking access methods specialized for media assets and fragments as well as to the variety of used description models. With *SPARQL-MM* we extend SPARQL, the standard query language for the Semantic Web with media specific concepts and functions to unify the access to Linked Media. In this paper we describe the motivation for *SPARQL-MM*, present the State of the Art of Linked Media description formats and Multimedia query languages, and outline the specification and implementation of the *SPARQL-MM* function set.

## Categories and Subject Descriptors

H.2.3 [**DATABASE MANAGEMENT**]: Languages; H.5.1 [**INFORMATION INTERFACES AND PRESENTATION**]: Multimedia Information Systems; H.5.4 [**INFORMATION INTERFACES AND PRESENTATION**]: Hypertext/Hypermedia

## Keywords

Linked Data, Multimedia, Query Language, Linked Media, SPARQL-MM

## 1. INTRODUCTION

One of the basic functions of a Database Management System is the efficient retrieval of stored data. The special needs of such a retrieval are strongly dependent on a) the stored data (and its underlying representation) and b) the specific use case. A proper retrieval mechanism for Multimedia in the Web of Data will be a mixture of classical multimedia functionalities and Semantic Web related data querying. Thus, we reformulate the standard definition for Information Retrieval in [23] as follows:

*Multimedia Retrieval on the Web of Data is finding (fragments of) resources of an unstructured nature (text, image, video, etc.) that satisfy an information need.*

whereby:

***Web of Data*** a dataspace of resources, which are represented in interchangeable, common formats, and interconnected by named links. Thus, the Web of Data is an exchange medium for data as well as documents, like described in the vision of the related W3C Data Activity group [33].

***finding*** means providing a subset of web resources that meets someones expectations and is human-manageable in presentation form and amount (ordered list, collage etc.).

***resources*** in this context all things that are addressable via common web standards. For a seamless integration of Linked Data principles [3], resources (information and non-information resources) must be accessible via HTTP protocol. In addition, the fragmentation of resources requires a suitable representation format, like e.g. the Media Fragments URI specification [31].

***unstructured nature*** a resource is not interpretable per se but must be interpreted by experts or specialized machines to extract common understandable structure and features.

***information need*** an abstract description of the expected subset or list. The more exact the information need is defined the more exact the presented set fits the expected results.

The query language can be seen as an instrument for formalizing this need. It is an interface between user needs and the (mostly abstract) multimedia data and metadata storage layer. In this paper we describe, how we adapt the SPARQL Protocol and RDF Query Language to support multimedia specific features.

*SPARQL-MM is developed within MICO (Media in Context), a research project partially funded by the European Commission 7th FP (grant agreement no: 610480).


*WWW 2015 Companion,* May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3473-0/15/05.
http://dx.doi.org/10.1145/2740908.2742914.

## 2. LINKED MEDIA

In 2007 the Linking Open Data (LOD) community project was initiated by the W3C [1]. The main goal was to bootstrap the Semantic Web by publishing datasets [7] following the design issues by Tim Berners-Lee [3] that outlined a best practice for exposing, sharing, and connecting pieces of data, information, and knowledge. Ideally it should enable the publication and interlinking of open data on the Semantic Web by identifying and using already existing sets of open data available on the World Wide Web, and of course by creating new linked datasets [4].

At the same time the amount of services dealing with multimedia data (Facebook, YouTube, Flickr, etc.) on the WWW is ever growing. Even if the description of digital media resources with metadata properties has a long history in research and industry [9] multimedia assets played a subsidiary role at the first steps of the Web of Data. In order to improve this situation, the W3C initiated the Video in the Web activity[2]. The associated Working Groups defined a media-format independent format for addressing media fragments on the Web using Uniform Resource Identifiers. This format supports particular name-value pairs, like `('t='start','end')` for temporal fragments and `('xywh=','x' ,'y','width','height')` for regional fragments. A further group developed a common description practice for many different media objects and formats on the web by providing an ontology [6] and API [2]. More complex ontologies that fulfill many higher-level requirements for media annotation like COMM[3] (more or less a re-engineering of MPEG-7 using DOLCE), M3O[4] or RICO are not widely accepted precisely because of their complexity [19], which is a big hurdle for web users and developers. A model which is not restricted to media annotation but about annotations on the Web in general is the Open Annotation Data Model (OADM) [27]. It allows the creation of annotations that are easily shareable between platforms while trying to satisfy complex requirements and being as easy as possible at the same time. Both the Ontology of Media Resources and the OADM support Media Fragment UIRs for fragment identification.

Even if there are many approaches to publish interlinked media data a well-suited solution for multimedia retrieval in the Semantic Web is lacking. The de-facto standard query language for RDF (SPARQL) [14] allows expressing discrete queries across diverse data sources, where the data is represented as RDF. It includes features like basic conjunctive patterns, value filters, optional patterns, and pattern disjunction. SPARQL is extendable in many ways and thus allows to add functionality that goes beyond the specification of either the SPARQL query language or the SPARQL protocol. They add problem specific functions like querying for trust [15], using similarity measures [18], etc. Currently there is no extension that brings multimedia specific features like spatio-temporal aspects or media similarity into SPARQL.

[1] `http://www.w3.org/wiki/SweoIG/TaskForces/CommunityProjects/LinkingOpenData`
[2] http://www.w3.org/2008/WebVideo/
[3] `http://comm.semanticweb.org/`
[4] `http://m3o.semantic-multimedia.org/ontology/2009/09/16/.`

## 3. CONCEPTS OF MULTIMEDIA QUERY LANGUAGES

The current landscape for Multimedia Query Languages is very broad and includes many different approaches. In this section, we categorize these approaches and highlight the features that enable a query language for multimedia retrieval. Existing multimedia query languages can be classified in 6 main categories that are:

1. languages that extend SQL as the common standard for querying relational databases or follow an SQL-like approach, like WebSSQL [34] or SQL/MM [25],

2. languages that build or extend query languages for object oriented databases like MOQL [21] or $\text{POQL}^{\text{MM}}$ [16],

3. languages that are focusing an XML metadata structure, like MMDOC-QL [22] or XQuery [5] (which is not explicitly build for Multimedia),

4. visual query languages, like MQuery [11] (that focus on visual timeline retrieval) or VisualMOQL [26],

5. approaches that allow query-by-example, like [17] or WS-QBE [28], and

6. languages that try to build a meta-language, which are metadata agnostic and thus can be shared/distributed over several storage backends, like MPQF [12].

The query languages described above support various multimedia specific features. Some of them try to cover all of them ([12]) and some are specialized on a specific one ( [28]). These features are:

**Query-by-keyword** specifies a pattern query on freetext fields. This query uses similarity metrics for string comparison (e.g. Levenshtein distance).

**Query-by-example** specifies a similarity or exact-match retrieval, whereby the query itself is a multimedia content (image, text etc.).

**Query-by-spatial-relationship** includes spatial relation like neighborhood (e.g. if object A is left beside object B) and/or spatial aggregation like bounding box within the query.

**Query-by-temporal-relationship** includes temporal relation like neighborhood (e.g. is object A appears after object B) and/or temporal aggregation like intermediate space.

**Query-by-relevance-feedback** specifies an iterative retrieval process that take into account the results of a previous search, which are rated as good or bad by users.

**Query-by-media-function** includes operations on media assets that add a higher level of semantics (face recognition function, concept detection, etc.) or functions that build fragments for further use (e.g. extract audio from a video item).

# 4. INTRODUCTION TO *SPARQL-MM*

In this section we introduce *SPARQL-MM* as an extension SPARQL, the de-facto standard query language for the Semantic Web. It aims to add multimedia specific features like described above. Currently *SPARQL-MM* supports spatio-temporal filter and aggregation functions.

## 4.1 Model

*SPARQL-MM* defines some core classes that are necessary to describe spatio-temporal properties as well as relational and aggregational functions. There are some vocabularies which could be used here, complex ones like MPEG7 [24] or very simple ones like Ninsuna [10]. With our ontology we tried to get the tradeoff between expressiveness and complexity. We provide mappings for such existing vocabularies to our model. In this paper we present some basic classes and properties in text that are necessary to describe the spatio-temporal functions we want to introduce. The ontology is currently in progress and might be adapted to upcoming issues. Figure 1 shows the main class model, a formal version can be found online[5].

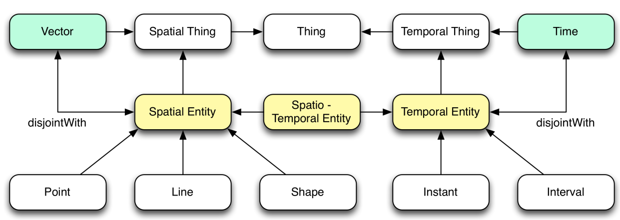


**Figure 1: SPARQL-MM basic classes and relations**

We introduce *mmo:SpatialEntity* and *mmo:TemporalEntity* to describe spatial as well as temporal instances and two classes to describe the actual values of the instances, whereby *mmo:Vector* is a superclass of all classes of multidimensional vectors and *mmo:Time* is a superclass of possible time representations (Normal Play Time NPT [29], etc.). As you can see, they are disjoint with each other.

**Spatial Entity** A superclass of any spatial entities like point, line, polygone, circle, etc.

**Temporal Entity** A superclass of any the temporal entity like instant and interval.

**Spatio Temporal Entity** A union class of spatio and temporal entities.

**Vector** A superclass for vectors.

**Time** A superclass for any kind of time specification.

The ontology describes several subclasses and properties for non abstract entities, like a class *mmo:Rectangle* with properties *hasXY* (describing the left-upper point) and *hasWH* (describing the dimension). It should be mentioned that we abstract from real units (e.g. percentage, pixel, etc), which makes the model more flexible for function definition. A mapping from existing models like the Media Fragments is straight forward: `http://example.org/image.jpg#xhwh=10,10,100,100` can be mapped to:

```
<> a mmo:Rectangle ;
  mmo:hasXY <v1> ;
  mmo:hasWH <v2> .
```

whereby `v1` and `v2` are vector resources that contain the associated values. Figure 2 outlines how *mmo:Rectangle* fits into the SPARQL-MM class model. The model is flexible and can be extended to more objects like complex shapes (e.g. defined with SVG) without major adaption.

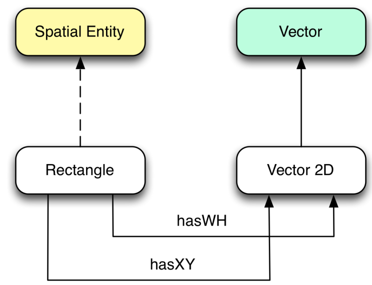


**Figure 2: A rectangular spatial object**

## 4.2 *SPARQL-MM Functions*

To provide the required functionality, we need to determine a model for spatial and temporal relations. Relations between spatial objects can be separated in three main classes, which are a) topological relations (relative relationship between spatial objects, like contains), b) directional relations (direction of spatial object $a$ in relation to spatial object $b$ e.g. rightBeside), and distance relations (the attributes of the relation itself, like nearby). In the following we briefly describe models for topological and directional relations as well as specific *SPARQL-MM* functions based on these models. Currently we do not consider distance relations, because they are fuzzy and therefore do not seamlessly integrate into SPARQL (unless extending SPARQL to fuzzy logic).

### *Topological Relations*

A standard model to describe relations between spatial objects in a 2 dimensional geometric model is the Dimensionally Extended nine-Intersection Model (DE-9im) [8]. The model is based on a 3x3 intersection matrix (Clementini-Matrix, Figure 3), which allows to specify the spatial relation of two geometric objects a and b according to interior (I), boundary (B) and exterior (E). The result of `dim(x)`

$$\begin{bmatrix} dim(I_a \cap I_b) & dim(I_a \cap B_b) & dim(I_a \cap E_b) \\ dim(B_a \cap I_b) & dim(B_a \cap B_b) & dim(B_a \cap E_b) \\ dim(E_a \cap I_b) & dim(E_a \cap B_b) & dim(E_a \cap E_b) \end{bmatrix}$$

**Figure 3: Clementini-Matrix**

is the maximum value of all matching intersection pattern, whereby `-1` is the value of $\emptyset$, `0` the dimension of point intersection, `1` the line intersection and `2` the dimension of area intersection. To get a compact string representation it is common to concatenate the pattern values from left-to-right and from top-to-bottom, e.g. `212101212`. As described in

[5] `https://raw.githubusercontent.com/tkurz/sparql-mm/master/ns/1.0.0/ontology/index.rdf`

[13] model allows us the specification of 10 spatial predicates with their patterns, which allows us to describe a topological function set for *SPARQL-MM*:

```
xsd:boolean mf:spatialEquals (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
xsd:boolean mf:disjoint (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
xsd:boolean mf:touches (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
xsd:boolean mf:spatialContains (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
xsd:boolean mf:covers (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
xsd:boolean mf:intersects (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
xsd:boolean mf:within (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
xsd:boolean mf:coveredBy (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
xsd:boolean mf:crosses (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
xsd:boolean mf:spatialOverlaps (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
```

Some of the names are overloaded, which means that they are used for temporal as well as spatial relations. Therefore we added a prefix *spatial* to some of them.

In addition to the functions we defined a set of topological aggregation functions. As we currently only consider rectangle shapes, we took a reasonable subset, which has to be extended in future work. This functions are:

```
lmo:SpatioTemporalEntity mf:boundingBox (
      lmo:SpatioTemporalEntity a,
      lmo:SpatioTemporalEntity b )
lmo:SpatioTemporalEntity mf:intersection (
      lmo:SpatioTemporalEntity a,
      lmo:SpatioTemporalEntity b )
```

The functions has *lmo:SpatioTemporalEntity* as input and output parameter, because they can be used for spatial as well as for temporal relations. It has to be mentioned that the implementation of the two aggregation functions does not have to be limited to two parameters but as both functions are associative, commutative and distributive they can be nested.

### Directional Relations

Like for topological relations we have to find predicates for directional relations to specify proper functions for *SPARQL-MM*. Directional relations describe how a primary object $a$ is placed relative to a reference object $b$ based on a coordinate system (for example, object $a$ is south of object $b$). There are several models, which describe directional relations in different spaces like described in [30]. Models like the projection-based or the cone-based model define relations between punctual objects but can be easily extended to spatial object by approximating an extended representative point (e.g. the centroid). Both models partition the space around the reference object $b$ into a number of mutually exclusive areas. Other models like PDR (Projection-based Directional Relation model) extend the definition to spatial objects which provides more preciseness and expressiveness but increases the number of relations that can be expressed (for PDR there are 511 possible relations), which disqualifies it as basis for directional predicates. We decided to take the projection-based model (which defines 8 relational functions) because a) it is easy to understand for users, b) it allows us to specify intuitive predicate names and c) it can be calculated very efficiently (by indexing the centroid for any spatial object). To make it even more intuitive, we refrain from using words from the geographical domain (*Geo*) and replaced it with daily used words (name of *name of Function*) as follows:

| Geo | Function |
|---|---|
| W | $left(a,b) = a.x < b.x$ |
| E | $right(a,b) = a.x > b.x$ |
| N | $above(a,b) = a.y > b.y$ |
| S | $below(a,b) = a.y < b.y$ |
| NW | $leftAbove(a,b) = leftBeside(a,b) \wedge above(a,b)$ |
| NE | $rightAbove(a,b) = rightBeside(a,b) \wedge above(a,b)$ |
| SW | $leftBelow(a,b) = leftBeside(a,b) \wedge below(a,b)$ |
| SE | $rightBelow(a,b) = rightBeside(a,b) \wedge below(a,b)$ |

Now we are able to describe a directional function set for *SPARQL-MM*:

```
xsd:boolean mf:leftBeside (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
xsd:boolean mf:rightBeside (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
xsd:boolean mf:above (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
xsd:boolean mf:below (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
xsd:boolean mf:leftAbove (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
xsd:boolean mf:rightAbove (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
xsd:boolean mf:leftBelow (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
xsd:boolean mf:rightBelow (
      lmo:SpatialEntity a, lmo:SpatialEntity b )
```

### Temporal Relations

The standard model for temporal relation was introduced by Allen's interval algebra for temporal reasoning [1]. The algebra defines thirteen basic relations between two time intervals, whereby a time point can be interpreted as a interval with duration 0. It has to be mentioned that 6 relations are converse (e.g. *precedes* p is converse to *preceded by* P). With this model we are able to describe a topological function set for *SPARQL-MM*:

```
xsd:boolean mf:precedes (
      lmo:TemporalEntity a, lmo:TemporalEntity b)
xsd:boolean mf:meets (
      lmo:TemporalEntity a, lmo:TemporalEntity b)
xsd:boolean mf:overlaps (
      lmo:TemporalEntity a, lmo:TemporalEntity b)
xsd:boolean mf:finishedBy (
      lmo:TemporalEntity a, lmo:TemporalEntity b)
xsd:boolean mf:contains (
      lmo:TemporalEntity a, lmo:TemporalEntity b)
```

```
xsd:boolean mf:starts (
      lmo:TemporalEntity a, lmo:TemporalEntity b)
xsd:boolean mf:equals (
      lmo:TemporalEntity a, lmo:TemporalEntity b)
xsd:boolean mf:startedBy (
      lmo:TemporalEntity a, lmo:TemporalEntity b)
xsd:boolean mf:during (
      lmo:TemporalEntity a, lmo:TemporalEntity b)
xsd:boolean mf:finishes (
      lmo:TemporalEntity a, lmo:TemporalEntity b)
xsd:boolean mf:overlapedBy (
      lmo:TemporalEntity a, lmo:TemporalEntity b)
xsd:boolean mf:metBy (
      lmo:TemporalEntity a, lmo:TemporalEntity b)
xsd:boolean mf:precededBy (
      lmo:TemporalEntity a, lmo:TemporalEntity b)
```

Two temporal relations are already described in the section about topological relations. We add one, which is specific to temporal relations only:

```
lmo:TemporalEntity mf:intermediate (
      lmo:TemporalEntity a, lmo:TemporalEntity b )
```

The current specification focuses on relational and aggregational functions. A support of temporal and spatial feature functions like *area*(*shape*) or *duration*(*interval*) will be considered in the next iteration of *SPARQL-MM*.

## 5. USING *SPARQL-MM*

Currently the implementation [20][6] supports Media Fragment URIs [31] for specifying spatial and temporal objects. We currently support all functions that are mentioned above but continuously extend the function set (adding spatio-temporal feature functions, allow complex shapes, etc.). The first iteration uses the OpenRDF Sesame[7] API and its extension interfaces, which makes it backend agnostic but requires expensive and inefficient in-memory calculations. It implements SPARQL-MM functions as *real* filters which are applied at the end of the SPARQL process. Thus it is also applicable for partitioned SPARQL evaluation like, e.g. used by Linked Media Fragments [32]. Currently we work on a backend specific implementation for Marmotta Triplestore[8] that allows query optimization by using inherent SPARQL-MM function characteristics.

Figure 4 outlines an example of an annotated video showing a hunting scene in the Serengeti. We use Media Fragment URIs to link annotations to specific spatio-temporal parts of the video. In our example a *wildebeest* appears from second `100` to `150` on the left side, while a *leopard* is marked from second `120` to `155` on the right side. If a user wants to retrieve the (spatio-temporal) snippet, that covers a specific hunting scene, she may issue a query like: "Give me the spatio-temporal snippet that shows a *big cat* right beside a *wildebeest*". Using *SPARQL-MM* functions we can now formulate this need as a SPARQL query (Listing 1). We use *mm:overlaps* to get fragments that appear in the identical temporal sequence. *mm:rightBeside* handles the spatial relation and *mm:boundingBox* merges every two fragments that match the filters. The result of the query is the media fragment `http://my.videos.org/v1.mp4#xywh=percent:20,30,80,45&t=120,150`.

[6]`https://github.com/tkurz/sparql-mm`

[7]`http://www.openrdf.org/`

[8]`http://marmotta.apache.org/kiwi/triplestore.html`

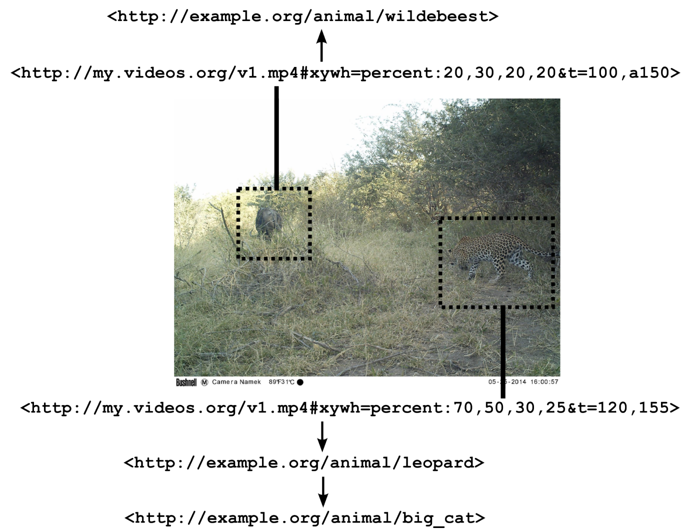


**Figure 4: A sample for an annotated video**

**Listing 1: A SPARQL-MM query**

```
SELECT (mf:boundingBox(?l1,?l2) AS ?hunter)
WHERE {
  ?f1 a ma:MediaFragment;
    ma:locator ?l1; dct:subject ?a1 .
  ?a1 skos:broader animal:big_cat .
  ?f2 a ma:MediaFragment; ma:locator ?l2 .
  ?f2 dct:subject animal:wildebeest .
  FILTER mf:rightBeside(?l1,?l2)
  FILTER mf:overlaps(?l1,?l2)
}
```

## 6. CONCLUSIONS AND FURTHER WORK

In this paper we introduced the SPARQL extension *SPARQL-MM*, which adds multimedia specific features to the standard query language for the Semantic Web. We introduced a simple class model which allows us to build ontology agnostic functions and thus support various data models describing spatio-temporal multimedia fragments. Currently we are extending *SPARQL-MM* with more feature functions and work on a more efficient implementation including benchmarking tests. In the future we will introduce similarity features that will allow to query annotated media by example.

# 7. REFERENCES


[1] J. Allen. Maintaining Knowledge About Temporal Intervals. *Communications of the ACM*, 26(11):832–843, 1983.

[2] W. Bailer, C. Poppe, W. Lee, M. Höffernig, and F. Stegmaier. Metadata API for media resources 1.0. W3C recommendation, W3C, Mar. 2014. http://www.w3.org/TR/2014/REC-mediaont-api-1.0-20140313/.

[3] T. Berners-Lee. Linked Data: Design Issues. http://www.w3.org/DesignIssues/LinkedData.html, 2006.

[4] C. Bizer, T. Heath, and u.a. Interlinking open data on the web, 2007. Stand 12.5.2009.

[5] S. Boag et al. XQuery 1.0: An XML Query Language. Technical report, W3C, 2007.

[6] P.-A. Champin et al. Ontology for media resources 1.0. W3C recommendation, W3C, Feb. 2012. http://www.w3.org/TR/2012/REC-mediaont-10-20120209/.

[7] T. B.-L. Christian Bizer, Tom Heath. Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems*, 5(3):1–22, 2009.

[8] E. Clementini, P. D. Felice, and P. van Oosterom. A small set of formal topological relationships suitable for end-user interaction. In D. J. Abel and B. C. Ooi, editors, *SSD*, volume 692 of *Lecture Notes in Computer Science*, pages 277–295. Springer, 1993.

[9] V. Damjanovic et al. Semantic Enhancement: The Key to Massive and Heterogeneous Data Pools. In *Proceeding of the 20th International IEEE ERK (Electrotechnical and Computer Science) Conference 2011, Portoroz, Slovenia*, September 2011.

[10] V. Deursen et al. Experiencing standardized media fragment annotations within html5. *Multimedia Tools and Applications*, 70(2):827–846, 2014.

[11] J. D. N. Dionisio and A. F. Cardenas. MQuery: A visual query language for multimedia, timeline and simulation data. *J Visual Languages and Computing*, 7:377–401, 1996.

[12] M. Döller, R. Tous, M. Gruhne, K. Yoon, M. Sano, and I. Burnett. The MPEG Query Format: Unifying Access to Multimedia Retrieval Systems. *IEEE Multimedia*, 15, 2008.

[13] M. J. Egenhofer and J. Herring. A mathematical framework for the definition of topological relationships. In *Fourth international symposium on spatial data handling*, pages 803–813. Zurich, Switzerland, 1990.

[14] S. Harris and A. Seaborne. SPARQL 1.1 Query Language, 2013.

[15] O. Hartig. Querying Trust in RDF Data with tSPARQL. In *6th Annual European Semantic Web Conference (ESWC2009)*, pages 5–20, June 2009.

[16] R. G. Henrich Andreas. POQLˆMM: A Query Language for Structured Multimedia Documents. *Proceedings 1st International Workshop on Multimedia Data and Document Engineering (MDDE'01)*, pages 22–229, 2001.

[17] Y. M. Jonathan Mamou. A Query Language for Multimedia Content. In *Proceeding of the Multimedia Infomation Retreival workshop*, 2007.

[18] C. Kiefer, A. Bernstein, and M. Stocker. The fundamentals of isparql - a virtual triple approach for similarity-based semantic web tasks. In *Proceedings of the 6th International Semantic Web Conference*, volume 4825, pages 295–308, November 2007.

[19] T. Kurz, G. Güntner, V. Damjanovic, S. Schaffert, and M. Fernandez. Semantic enhancement for media asset management systems. *Multimedia Tools and Applications*, pages 1–27, 2012. 10.1007/s11042-012-1197-7.

[20] T. Kurz, S. Schaffert, K. Schlegel, F. Stegmaier, and H. Kosch. SPARQL-MM-Extending SPARQL to Media Fragments. *The Semantic Web: ESWC 2014 Satellite Events*, pages 236–240, 2014.

[21] J. Z. Li et al. MOQL: A Multimedia Object Query Language. In *in Proceedings of the 3RD international workshop on Multimedia Information System*, 1997.

[22] P. Liu, A. Chakraborty, and L. H. Hsu. A Logic Approach for MPEG-7 XML Document Queries. In *Proceedings of Extreme Markup Languages®*, 2001.

[23] C. D. Manning and P. Raghavan. *An Introduction to Information Retrieval*. Cambridge University Press, 2009.

[24] J. Martinez, R. Koenen, and F. Pereira. MPEG-7: The Generic Multimedia Content Description Standard, part 1. *IEEE Multimedia*, 9, 2002.

[25] J. Melton and A. Eisenberg. SQL Multimedia and Application Packages (SQL/MM). *SIGMOD Rec.*, 30(4):97–102, Dec. 2001.

[26] V. Oria et al. VisualMOQL: the DISIMA visual query language. *Proceedings IEEE International Conference on Multimedia Computing and Systems*, 1, 1999.

[27] R. Sanderson, P. Ciccarese, and H. V. de Sompel. Open Annotation Data Model. Community Draft, Open Annotation Collaboration, Feb. 2013. http://www.openannotation.org/spec/core/.

[28] I. Schmitt, N. Schulz, and T. Herstel. WS-QBE: A QBE-Like Query Language for Complex Multimedia Queries. *11th International Multimedia Modelling Conference*, 2005.

[29] H. Schulzrinne, A. Rao, and R. Lanphier. RFC2326 - Real Time Streaming Protocol (RTSP), April 1998.

[30] S. Skiadopoulos et al. A family of directional relation models for extended objects. *Knowledge and Data Engineering, IEEE Transactions on*, 19(8):1116–1130, 2007.

[31] R. Troncy, D. V. Deursen, E. Mannens, and S. Pfeiffer. Media Fragments URI 1.0 (basic). W3C recommendation, W3C, Sept. 2012.

[32] R. Verborgh et al. Web-scale querying through linked data fragments. In *Proceedings of the 7th Workshop on Linked Data on the Web, LDOW14*, 2014.

[33] W3C. W3C Data Activity. http://www.w3.org/2013/data/, 2013.

[34] C. Zhang, W. Meng, Z. Wu, and Z. Zhang. WebSSQL - A Query Language for Multimedia Web Documents. In M. P. Papazoglou and A. Sheth, editors, *IEEE Advances in Digital Libraries 2000 - ADL2000*, page 10. IEEE Computer Society, 2000.