OpenKnowledge

FP6-027253

Visualisation Markup Tools

Antonis Loizou, Mischa M. Tuffield, Paul Lewis, David Dupplaw,
Madalina Croitoru, Liang Xiao, and Srinandan Dasmahapatra

School of Electronics and Computer Science, University of Southampton, UK

Report Version: final
Report Preparation Date: 23.7.2007
Classification: deliverable D5.2
Contract Start Date: 1.1.2006 Duration: 36 months
Project Co-ordinator: University of Edinburgh (David Robertson)
Partners: IIIA(CSIC) Barcelona
Vrije Universiteit Amsterdam
University of Edinburgh
KMI, Open University
University of Southampton
University of Trento

Markup Tools for OpenKnowledge

Antonis Loizou, Mischa M. Tuffield, Paul H. Lewis, David Dupplaw, Madalina Croitoru, Liang Xiao, Srinandan Dasmahapatra

July 23, 2007

Abstract

The OpenKnowledge system needs to be able to deal with all kinds of data and to that end we cannot impose any formal data structures on the variables that occur within the interaction models. However, we can show that using a common format for some operations will make the system more interoperable without the need for creating translation services. We show here that using a single translation service that can transfer between an MPEG-7 representation, we can provide some multimedia manipulation services that when combined provide complex applications built in LCC.

1 Introduction and Motivation

The aspiration of OpenKnowledge is to allow knowledge to be shared freely and reliably, regardless of the source or the consumer. Reliability here is interpreted as a semantic issue, as the Internet is in the fortunate position that transport-level reliability has already been established. Reliably sharing semantic data requires either a shared conceptualisation where there is consensus, or the mapping of semantic terms between locally defined models. However, the supplier and consumer of such knowledge need a context in which mapping can be performed, and models of interaction can provide this context. It is for this reason that OpenKnowledge has its core mechanism in the sharing of *interaction models*, and it is expected that communities can form in which the sharing of knowledge mappings can be exploited.

A problem arises when data interactions require data that is not easily annotated. Current methods used to extract semantics from image, audio, or video (multimedia) data are not as mature as their text-based counterparts and are presented as the test-bed for this Work Package.

Work Package 5 addresses these aspects by showing how content-based multimedia annotation techniques can be utilised in the OpenKnowledge environment.

Very simply, the main objectives of the work package are:

- To provide mechanisms to support the incorporation of multimedia items as components of the OpenKnowledge system.

- To provide forms of visualisation that allow interaction with and monitoring of interaction models, mappings and routing.

Deliverable 5.1 worked towards the second objective. In that deliverable we described mechanisms for coupling the interaction models to data visualisation implementations by a simple mark-up language in the LCC [25] (Lightweight Coordination Calculus) model. The mark-up language mapped constraints in the interaction models to *visual terms* that described a type of visual interaction with the user. The implementation of any specific visual term was reliant on the specific plug-in that the user had installed for that type of visualisation, which provided a means for extending the visualization system and for customisation. The OpenKnowledge kernel now has an initial visualisation system that shows the technique is workable.

Having visualisation of a data item that originates from placeholders in the interaction model assumes that the data is already annotated, or that the visualiser completely understands the incoming data. So, a natural step forward from the visualisation is to provide semantic handles for the multimedia providing the same interoperability that textual data enjoys including ontology mapping.

In this deliverable we will identify and address the extra challenges posed by multimedia data annotation and the benefits of an "open" approach to multimedia data manipulation. Our purpose is to be able to exchange multimedia data in a flexible, reusable manner aimed at addressing the "semantic gap" - the differential between the information that can be extracted automatically from a digitised image and the interpretation that humans might place upon the image. We show that a flexible approach, as provided by OpenKnowledge, will have clear benefits for building multimedia applications by easing multimedia data exchange. This deliverable will expand on this claim and introduce a scenario in which we can demonstrate the claim.

As a concrete outcome this deliverable will describe a tool that provides, in a semi-automatic way, low to high level mappings for multimedia data. These mappings will be obtained using Interaction Models that could then be reused for different sets of mappings. We thus show, on one hand the OpenKnowledge project's ability for handling multimedia data and, on the other, specific advantages that arise from addressing multimedia in an OpenKnowledge scenario.

2 Background

The OpenKnowledge system needs to be able to deal with all kinds of data and to that end we cannot impose any formal data structures on the variables that occur within the interaction models. However, we can show that using a standardised framework for certain data will increase the system's interoperability without the need for creating multiple translation services or relying on ontology mapping for schema-level interoperability. There are few standardised metadata formats for multimedia, but the most comprehensive is that of MPEG-7.

2.1 MPEG-7

MPEG-7[1] is a standard developed by the Moving Pictures Expert Group (MPEG) to aid in the representation of multimedia data and metadata. The motivation for MPEG-7 (formally called the "Multimedia Content Description Interface") is to provide a standardised means for representing content and metadata for multimedia to aid in producing interoperable indexing systems for non-text media. MPEG-7 is of interest in the OpenKnowledge network because it provides a way to annotate media regardless of how that media is stored or delivered.

MPEG-7 standardises the definition of *Descriptors* that are used to provide information about media files, including text annotations, content-based annotations and non-formal semantic annotations. Sets of *Descriptors* can be related to each other using *DescriptorSchemes*, that provide non-formalised relationships between the *Descriptors*. MPEG-7 defines a language (the *Descriptor Definition Language* or DDL) that is used to create these structures. The DDL was initially defined using the XML-Schema but has since been formalised in the creation of a Semantic Web representation of the standard. Instances marked up against the MPEG-7 ontology could then be mapped to other instance data using the ontology mapping techniques being developed in OpenKnowledge.

RDF (The Resource Description Framework) is the standard language of the Semantic Web because it provides a means for defining classes and their relationships simply. There have been some concerted attempts to convert the MPEG-7 schema into an ontology, including Jane Hunter's [14] and Chirsa Tsinaraki's [6] work. Roberto García's work [10] has provided an automatic and complete translation of the definition into an OWL-Full ontology containing over 2300 classes. Other similar work includes Raphael Troncy's work [29], who considered the MPEG-7 data schema to be too cumbersome and defined his own extensible language for multimedia based on MPEG-7.

For use in OpenKnowledge we have compiled an MPEG-7 library that can be used within OKCs (Open Knowledge Components) for generating MPEG-7 data. OKCs that require this functionality can either use it directly as a library in the OKC implementation, or extend a special OKC that provides some functionality for dealing with MPEG-7 data.

2.2 OpenKnowledge Multimedia Example

As a simple example to illustrate how MPEG-7 can be used in the context of OpenKnowledge, first we will present an image-retrieval interaction model, and the associated OpenKnowledge component that will retrieve the image. This defines the fundamental interactions that will be required for all image analysis models that will require image data. The MPEG-7 library is used to extract certain values into the interaction model state such that the relevant values can be passed around.

[1]MPEG-7 *http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm*

$$
\begin{array}{l}
\mathbf{a}(\text{image\_searcher}, ID1) :: \\
\quad keyword(Z) \ \Rightarrow \ \mathbf{a}(\text{image\_provider}, ID2) \leftarrow \mathbf{getKeyword}(Z)\ then \\
\quad (\\
\qquad image(I) \ \Leftarrow \ \mathbf{a}(\text{image\_provider}, ID2) \leftarrow \mathbf{show}(I) \\
\qquad or \\
\qquad (not\_available(K) \ \Leftarrow \ \mathbf{a}(\text{image\_provider}, ID2) \leftarrow \mathbf{error}(K)\ then \\
\qquad \mathbf{a}(\text{image\_searcher}, ID1)) \\
\quad) \\
\\
\mathbf{visual}(\mathbf{getKeyword}(Z), \mathbf{input}(Z)) \\
\mathbf{visual}(\mathbf{show}(I), \mathbf{image}(I)) \\
\mathbf{visual}(\mathbf{error}(K), \mathbf{text}(\text{“No image for keyword:”}, K))
\end{array}
\tag{1}
$$

Model 1 shows a model for a role that searches an *image_provider* role for images matching a certain keyword. It sends a keyword, Z, to the image provider who will return either a *not_available* message, indicating that there were no images that matched the given keyword, or an *image* message containing information about the image in the variable I. The variable I will not be the binary representation of the image, as this may be large in size and would significantly slow down the interaction model processing, sometime unnecessarily, so the variable I will contain an MPEG-7 document that contains both intentional and automatically generated metadata about the image. This may include, for example, the URL of the image on the internet, or some other description of how to retrieve the image, and the visual term, *show*, that is mapped to the visualisation *image*, will be responsible, in this case, for retrieving that image. The library we provide can assist in retrieving the binary form of images for certain protocols, like HTTP.

The variable I in Model 1 is used to transfer metadata about the image. As OpenKnowledge does not enforce any specific data formats upon its variable values, we can use MPEG-7 for sending this data, because MPEG-7 provides a standardised, extensible framework for multimedia metadata. The MPEG-7 standard defines the representation in an XML schema, and Figure 1 shows an excerpt from an MPEG-7 document that can be transferred in the LCC variable. The figure shows how low-level information, such as the camera's EXIF metadata, can be transferred in MPEG-7, as well as high-level features, such as the information about the creator of the image. Other information such as the low-level features from the image that have been extracted automatically can also be provided in the MPEG-7 description, such as colour information in histograms, or edge information in vectors. MPEG-7 also provides a means for adding semantic tags to the image, and with semi-automatic mapping tools that are described later, these can be popualted.

```xml
<?xml version="1.0" encoding="UTF-8" ?>
  <Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001"
     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <CreationInformation>
      <Creation>
        <Title>Creation information</Title>
          <Creator>
            <Role href="creatorCS">
              <Name>Creator</Name>
            </Role>
            <Agent xsi:type="PersonType">
              <Name>
                <GivenName>David</GivenName>
                <FamilyName>Dupplaw</FamilyName>
              </Name>
              <ElectronicAddress>
                <Url>http://david.dupplaw.me.uk/</Url>
              </ElectronicAddress>
            </Agent>
          </Creator>
          <CreationCoordinates>
            <Date>
              <TimePoint>2005-05-18T08:32:09</TimePoint>
            </Date>
          </CreationCoordinates>
          <CreationTool>
            <Tool>
              <Name>Canon Canon PowerShot S45</Name>
            </Tool>
            <Setting name="Make" value="Canon" />
            <Setting name="Model" value="Canon PowerShot S45" />
            <Setting name="Orientation"
                value="Top, left side (Horizontal / normal)" />
            <Setting name="X Resolution" value="180 dots per inch" />
            <Setting name="Y Resolution" value="180 dots per inch" />
            <Setting name="Resolution Unit" value="Inch" />
            <Setting name="Date/Time" value="2005:05:18 08:32:09" />
            <Setting name="YCbCr Positioning"
                value="Center of pixel array" />
            <Setting name="Exposure Time" value="1/640 sec" />
            <Setting name="F-Number" value="F2.8" />
                  ...
```

Figure 1: An MPEG-7 XML fragment for an image

$$
\begin{array}{l}
\mathbf{a}(\text{image\_provider}, ID2) :: \\
\quad keyword(Z) \Leftarrow \mathbf{a}(\text{image\_searcher}, ID1) \; then \\
\quad (\\
\quad\quad image(I) \Rightarrow \mathbf{a}(\text{image\_searcher}, ID1) \leftarrow \mathbf{available}(K, I) \\
\quad\quad or \\
\quad\quad not\_available(K) \Rightarrow \mathbf{a}(\text{image\_searcher}, ID1) \\
\quad)
\end{array}
\tag{2}
$$

Model 2 shows the image provider role that allows a peer to share images on the network. Clearly, the image provider needs to be able to produce MPEG-7 documents with which to populate the variable I in the model, and it needs to do this in the constraint **available**(K, I). The constraints that are defined in the role are supported by the MPEG-7 OpenKnowledge component from which the image provider is able to extend, providing maximum reuse and minimum authoring overhead. For example, Listing 1 shows the Java implementation of the OKC for the MPEG-7 image provider. It not only demonstrates the ease of programming OKCs in general, but that by providing an MPEG-7 library implemented as an abstract OKC class, it minimises the overhead for creating MPEG-7-compatible OKCs. The method *getMpeg7Descriptor* is defined in the superclass and provides an XML document for a given image, like that shown in Figure 1.

2.3 Content-Based Image Annotation

The semantics of textual media can be extracted using natural language techniques. These use the explicit semantics of the atomic parts of a sentence (the words) to build more generalized semantic annotations for text. However, determining the semantics of atomic parts of multimedia data is a much more difficult task, because the semantics are not explicit as they are for words and the atoms are very hard to identify. For example, images may contain many millions of pixels, some of which when grouped together form a representation of a specific semantic concept, for example, a car. However, those pixels are subject to large amounts of variation, even in similar representations.

Automatic annotation attempts to bridge the gap between low level descriptors and symbolic labels effectively by learning mappings between combinations of low level media descriptors representing objects, and textual labels indicating the identity of those objects. One of the early attempts at automatic image annotation was the work of Mori et al [24] which attempted to apply a co-occurrence model to keywords and low-level features of rectangular image regions. Current techniques for auto-annotation generally fall into two categories; those that first segment images into regions, or 'blobs' and those that take a more scene-orientated approach, using global information.

The segmentation approach has recently been adopted by a number of researchers. Duygulu et al [8] proposed a machine translation approach to translate between keyword annotations and a discrete vocabulary of clustered 'blobs'.

```java
package org.openk.okc.multimedia;
import org.openk.okc.multimedia.Mpeg7OKC;

public class ImageProviderOKC extends Mpeg7OKC
{
  /**
   *  Implementation of the available(K,I) constraint
   */
  public void available(Argument Keyword, Argument Image)
  {
    // Get images for the given keyword
    URL[] imageList = getImages(Keyword.getValue());

    // Choose first image & create basic MPEG-7
      descriptor
    Image.setValue(getMpeg7Descriptor(imageList[0]));
  }

  private URL[] getImages(String keyword)
  {
    ...
  }
}
```

Listing 1: OpenKnowledge Component extending the MPEG-7 OKC

The data-set proposed by Duygulu et al [8] has become a popular benchmark of annotation systems in the literature. Jeon et al [15] improved on the results of Duygulu et al [8] by recasting the problem as cross-lingual information retrieval and applying the Cross-Media Relevance Model (CMRM) to the annotation task. Lavrenko et al [17] used the Continuous-space Relevance Model (CRM) to build continuous probability density functions to describe the process of generating blob features. The CRM model was shown to outperform the CMRM model significantly. The models by Monay and Gatica-Perez, [23] Feng et al [9] and Jeon and Manmatha [16] use rectangular regions rather than blobs. Monay and Gatica-Perez [23] investigate Latent Space models of annotation using Latent Semantic Analysis and Probabilistic Latent Semantic Analysis, Feng et al [9] use a multiple Bernoulli distribution to model the relationship between the blocks and keywords, whilst Jeon and Manmatha [16] use a machine translation approach based on Maximum Entropy.

In our own work in Southampton Hare and Lewis [11] showed how vector-space representations of image content, created from local descriptors of salient regions within an image, can be used for auto-annotation by propagating semantics from similar images. In Tang, Hare and Lewis [28] we introduced a

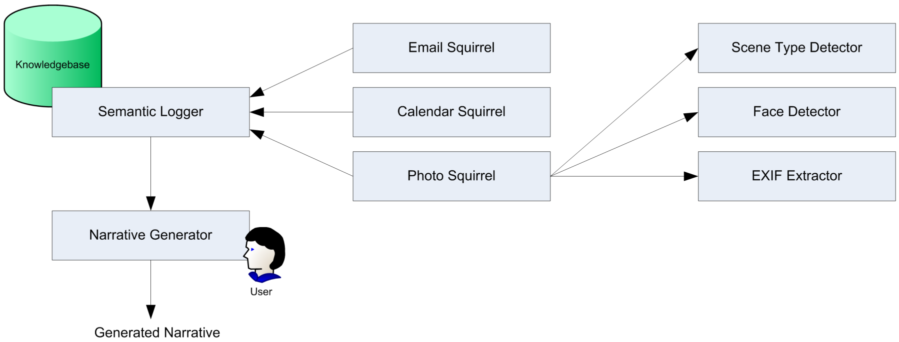

Figure 2: The scenario using multimedia annotations via a knowledge–base to generate narratives

statistical model using salient regions for auto image annotation and compared it with several earlier techniques.

Most of the auto-annotation approaches described above perform annotations in a hard manner; that is, they explicitly generate and assign a number of annotations to an image. In Hare et al [12] we introduce an approach in which annotation is performed implicitly in a soft manner. We create a multidimensional semantic space into which textual terms, visual terms and annotated and unannotated images are mapped in such a way that semantically similar entities are located near to each other. Images may then be retrieved either by textual descriptors or visual descriptors

When annotations are available for media objects, the annotations can be added to an MPEG-7 descriptor of that object, allowing the annotations to be understood by other analysis tools, or in the case of OpenKnowledge, other OKCs that understand MPEG-7.

3 Scenarios and Mapping Tools

We introduce a scenario that incorporates an illustration of a mechanism for managing multimedia artifacts in the OpenKnowledge system. The scenario is based upon on going research into the acquisition and exploitation of auto–biographical metadata. In this scenario the data is gathered inconspicuously and is, subsequently, automatically annotated to support a number of context-based services. The system described involves a number of data gathering processes, known as propagators, that periodically capture and import information about a user into their knowledge-base. The aim is to use annotations that have been gathered automatically from multiple data sources to provide a narrative of a user's actions relating to some specific occurrence. The overview of the system's core components is shown in Figure 2.

3.1 Semantic Logger

The Semantic Logger[2] (SL) is an auto-biographical metadata acquisition system that has been presented as a means to populate the Semantic Web with personal metadata. It has been developed as a platform for investigations into the utility of context aware systems, and is being ported to the OpenKnowledge system to aid in demonstrating the robustness of the architecture, with respect to handling multimedia. Extensive details of these projects can be found in [32, 19, 20, 30].

3.1.1 System Overview

The system builds on the ideas brought forward in the original *Scientific American* Semantic Web article [4], with a particular focus on the notion of assembling, and integrating web accessible resources. At his keynote speech during the International Semantic Web Conference 2003 [3] Tim Berners-Lee identified the *'Killer App for the Semantic Web'*, not as a single application but the successful integration of information, or to use his blunt words, *'Its the integration, stupid!'*.

The Semantic Squirrel Special Interest Group (SSSIG) [3] is a group of researchers based at the University of Southampton who aim to automate the process of logging any available raw data, (or *'nuts'*), that describe aspects of one's personal experience.

This raw data forms the basis of the knowledge acquisition phase for the Semantic Logger. Sources of contextual information are identified, captured, and parsed into RDF representations. The intent is to utilise this raw data about people to construct the context of a particular event, at a particular time. By virtue of the fact that each event logged by the system is time-stamped and related to the user's FOAF[4] URI[5], we are able to choose variable levels of granularity to describe any given context.

Upon registration of a Semantic Logger account, a user is presented with tools for the unobtrusive recording of personal information. The set of information sources presented in figure 3 is far from an exhaustive one, and is not intended to limit the functionality of the system. SL has been designed in a manner to allow information, in various forms of RDF to be posted to the knowledge base (KB).

At the heart of the system is the AKT Project's [6] SPARQL-compliant RDF triplestore 3store [13]. The key role of the triplestore is to act as a persistent store for the system, and to mediate the interactions between the other system components.

The system has a service-based architecture, and has been designed so that new services may join on an ad-hoc basis. The interactions with the central RDF triplestore make use of the SPARQL RDF query language [33]. The system uses

[2] Semantic Logger, http://akt.ecs.soton.ac.uk:8080/
[3] http://www.semantic-squirrel.org/
[4] Friend of a Friend (FOAF) *http://www.foaf-project.org*
[5] Uniform Resource Identifier *http://www.w3.org/Addressing/*
[6] Advanced Knowledge Technologies, http://www.aktors.org/

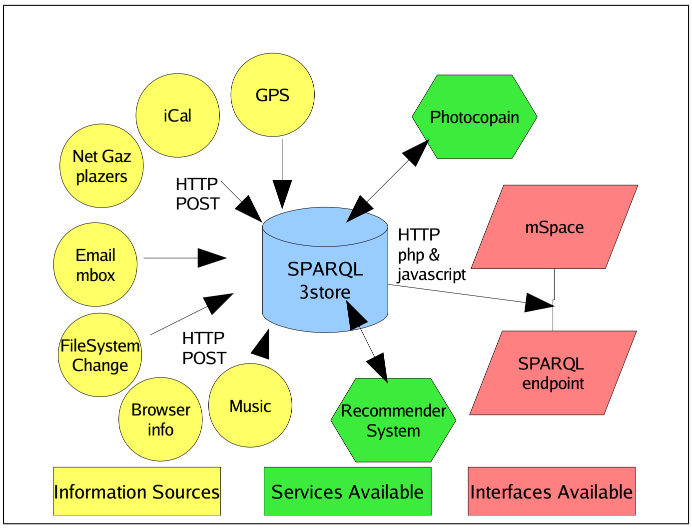

Figure 3: Overview of the Semantic Logger architecture, taken from [[32]]

a Universal Resource Identifier (URI) to point to a user's FOAF file and each user's FOAF file is linked to their RDF data. This URI is subsequently used to log the provenance of all the information asserted in the Semantic Logger.

The main requirements in selecting an appropriate RDF Knowledge Base implementation were efficiency and consistency. 3store is a system benchmarked against other RDF storage and query engines such as Jena [22], Sesame [5] and Parka [26] and shown to outperform all others in terms of both efficiency and scalability [27, 18].

3.1.2 Sources of Contextual Information

The SL collects, and propagates the forms of information into RDF representations described below. Effort has been put in selecting appropriate representations. They have been taken from proposed standards at the W3C or other standard making bodies, or have been selected due to current uptake on the web. Where such standards have not been available, we constructed local ontologies which describe the given phenomenon[7], while maintaining simplicity and generality. The metadata sources listed below are the foundations for the autobiographical contextual log captured:

[7] http://akt.ecs.soton.ac.uk:8080/downloads.php

- Calendar entries

  The SL has adopted the W3C recommendation in representing calendar entries in RDF [8]. A client-side application is available for download from the Semantic Logger site to automate the export of iCal [7] files (commonly used and platform independent) into this representation. In addition to querying capabilities as before, calendar entries can serve as context indicators for geographical locations (described below), enabling to an extent the resolution of co-location.

- Geo-Data

  In an attempt to build up a log of a user's geographical data, a two pronged approach is taken. For research purposes we have been carrying around GPS units to log our data. This information is extracted and parsed into an RDF representation, taken from *http://www.hackdiary.com/*. The RDF model builds on-top of the dublin core namespace[9], and W3C's recommendation for geographical data[10].

  GPS information is being used to track a user's change of location, but is not always a suitable method of tracking, for tall buildings, and movement between buildings within close proximity is hard to track, so a decision was taken to start employing a network gazetteer. The network gazetteer from Plazes is currently being employed by the Semantic Logger. Plazes[11] supplies the end user with client side applications that pick up a laptop's current network connection and provides information about the location if information has been entered for that WiFi[12] hotspot. Plazes provides a comprehensive API, and RSS 1.0 feeds, that export parsable RDF of a users activity. A decision has been taken to adopt their namespace for the purposes of logging network activity.

  The combination of the GPS information, a user's network gazetteer (given that the user has a laptop computer), and a his/her iCal file, along with the Getty Geographical Name gazetteer[13], allow us to infer a user's geographical context.

- Music playcount statistics

  Audioscrobbler[14], is a music search engine based on a large collection of music profiles. These profiles are built through allowing the users to download and install plugins to their respective media players that propagate the information to the system. The representation used to describe artists is Musicbrainz[15], a freely accessible knowledge base for the music

[8] iCal RDF representation: *http://www.w3.org/2002/12/cal/ical*
[9] Dublin Core namespace: *http://dublincore.org/documents/dces*
[10] W3C Geo-Data Namespace: *http://www.w3.org/2003/01/geo/*
[11] Plazes: *http://www.plazes.com/*
[12] Wireless Internet Network
[13] TGN: *http://www.getty.edu/research/conducting_research/vocabularies/tgn/*
[14] Audioscrobbler: *http://www.audioscrobbler.net*
[15] Musicbrainz: *http://musicbrainz.org/mm/mm-2.1*

domain, that publishes the data in their ontology in an attempt to provide a comprehensive music information service. These systems are currently in the process of developing their metadata vocabulary to be published in an ontology. In this interim phase, a local version is being developed to describe the data made available through their web-service API.

- Firefox bookmarks, downloads and navigation history

  By virtue of its cross-platform nature, the Mozilla Firefox has been selected as our web-browser of choice. Firefox exposes the download information in RDF form[16] and thus can be easily imported to the system. Scripts have been developed to parse the bookmarks and history data into RDF. The RDF model uses two namespaces taken from the Mozilla developers centre [17].

- Email

  A simple ontology has been constructed to describe email correspondence[18] as one of satisfactory quality has not been found to be readily available. A client-side application has been developed to parse and convert the widely used MBOX representation into the local format. The intended use of this information, in addition to the ability to query one's records, is to support the identification of communities of practice, under a predefined temporal context.

- File System Information

  Beagle[19] search indexes every file found on a user's computer. This is achieved by combining specialised analysis tools for extracting content from different file types. This creates a personal information space describing a computer at the file-system level. The information is parsed into a simple ontology and can be loaded into a user's SL. This enables services to detect the presence and usage of files, giving an indication to a user's interests.

- Weather Information

  Weather information is proposed as a means of putting an event into context. The weather service that captures weather information given a time and location is still under development. Work is underway to harvest data from *www.weather.com*.

- News Headlines

  The capturing of News Headlines is also presented as another method of enriching personal context. Work is underway to harvest New Headlines

[16]Netscape namespace: *http://home.netscape.com/NC-rdf#*
[17]Mozilla namespace: *http://developer.mozilla.org/en/docs/XUL_Tutorial:_RDF_Datasources*
[18]Semantic Logger email namespace: http://semanticlogger.ecs.soton.ac.uk/email/#
[19]The Beagle Project: *http://beagle-project.org*

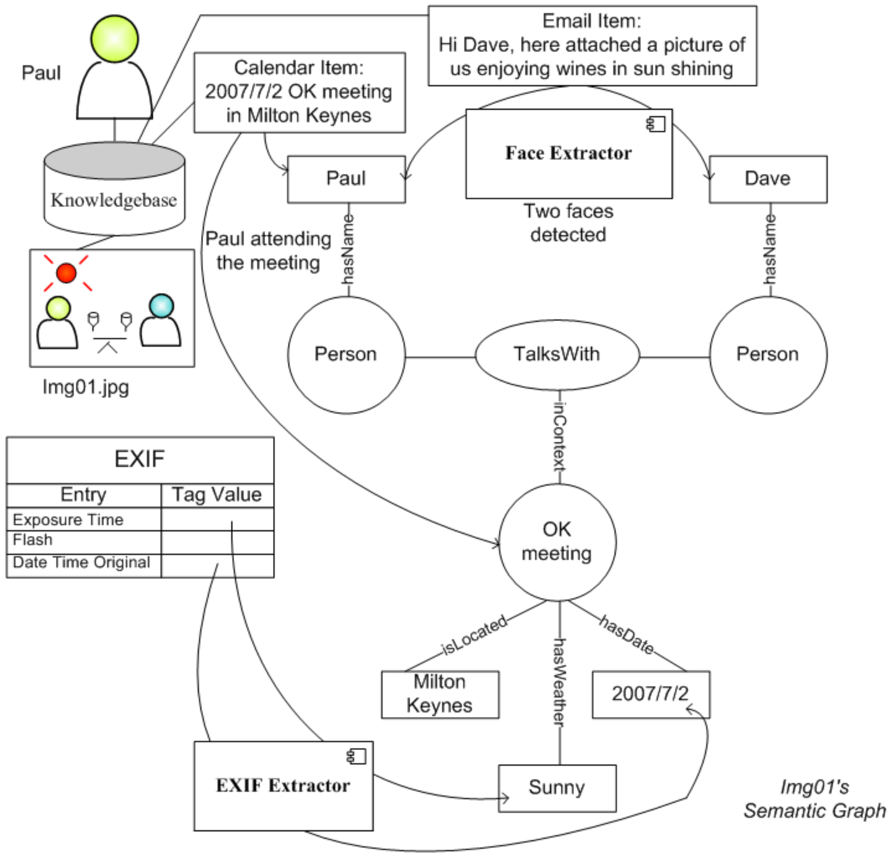

Figure 4: The scenario using multimedia annotations for Semantic Graphs

from the BBC website[20]. This is presented as another means of placing events into context.

3.2 A Scenario with Semantic Graphs

As mentioned before, the semantic gap between content-based image features and concept-based image descriptions is a major barrier to the understanding and processing of multimedia data. MPEG-7 specifies Descriptors (D) and Descriptor Schemes (DS) for the definition of conceptual objects and their relationships in the multimedia content (i.e. Person A talks with Person B). These can be used for constructing semantic graphs. We use a real life scenario to illustrate the automatic construction of semantic graphs by using various squirrels (as shown in Figure 2), a step further to the generation of narratives.

[20]BBC News website: *http://www.bbc.co.uk/news/*

Suppose a picture is taken in which Paul is talking with Dave about an OpenKnowledge meeting and the picture is stored in Paul's knowledgebase. The EXIF extraction component is invoked to retrieve the metadata that the camera has attached to the picture. This reveals the picture was taken on, say, 2nd July 2007, and by inference on this data, it can be determined that it was taken on a sunny day. The data from the Calendar Squirrel allows us to search for diary items added on that day and at that time by Paul and an item is found for an OpenKnowledge meeting in Milton Keynes which Paul plans to attend (high level semantic descriptions). Having aggregated this knowledge it can be inferred that this picture is about Paul attending the meeting. Furthermore, the data from the Email Squirrel shows an email that Paul sent to Dave with the picture attached (high level semantic descriptions) and the Face Detector finds two human faces in the picture (low level image features). As a result, a fact is inferred that the picture has both Paul and Dave on it, further supported by the previous semantic descriptions. A conversation between Paul and Dave could be explicitly stated in the email, but this could also be inferred by ontological reasoning. Alternatively, simple keywords/descriptions such as "talk, Dave" may be supplied by Paul interactively and associated with the picture. Eventually, through the aggregation of the RDF data, a semantic graph is built-up with knowledge including both the low level image features and high level semantic descriptions used in combination, as shown in Figure 4. Such graphs are extendable since, when additional knowledge is accumulated and connected with the existing knowledge presented in the graph, the meaning of the graph becomes richer. For example, the sample scenario may be extended to express that Paul talks with Dave on planning the future work of multimedia annotation assuming a squirrel could extract that information.

The knowledge representation in semantic graphs not only visualises the multimedia content for human understanding but also facilitates retrieval. When images are being queried the semantic graphs can be used to rank the relevance of images against the queries, where all the conceptual objects and their relationships captured in the graphs are compared with the ones required in the query. The preciseness of query results can also be judged according to the ontological relationships of concepts.

In conclusion, the complementary natures of low level image features (extractable and detectable by various modules shown in Figure 1) and high level semantic descriptions (collectable by various squirrels shown in Figure 1) are exploited in the semantic graph and it is demonstrated that, if they are combined in use for multimedia annotations they provide better expressiveness.

3.2.1 Summary

Ideally, all digital information created by a user would be encapsulated in the system. The user should have complete control over the information that will be processed. This can be ensured by providing each user with a distinct knowledge base to which only themselves, and services they subscribe have access. Furthermore, users are allowed to select which aspects of their personal experience they would like the system to capture.

It is expected that even the minimal burden of registering and configuring the system, coupled with reasonable privacy concerns can still be off–putting to some users. Joining the system can be made to appear more attractive, by supporting the users in extracting immediate added value from the information they have exported. As an effect of integrating this information into a single knowledge base, much more complex queries may be resolved (e.g. how many hyperlinks have I received in email correspondence and not yet visited?). In addition, by carrying out inference over this knowledge base, events that have occurred at the same place or time (or any other attribute) can be identified, allowing the information to be indexed based on such attributes.

It can be argued that immediate added value emerges from the use of this system. Firstly, support for this argument arises from enabling the application of SPARQL[21] queries on the available information, to answer questions that would be unfeasible under representations of singular domains. For example, named entity recognition can be applied to email correspondence to identify closely related groups while co-authorship and co-reference between scholarly articles can be analysed as shown in [2]. Co-location at various events can be inferred from geo-data and calendar entries, while the latter, in combination with the analysis of locally stored multimedia files (e.g. music and video files) can aid in identifying events of interests.

In addition since information is represented in an RDF graph, by virtue of the representation there exist multiple dimensions in which the data may be indexed and viewed. The mSpace interface[21] has the ability to organise such data, in multi–pane browsers. Furthermore, the edges of the graph are allowed to be reordered using dimensional sorting, independent of the hierarchical nature of the representation, allowing for a number of such trees to be visualised and browsed. mSpace requires the definition of a *default column* and a *target column* along with the path, through ontological relationships (edges in the graph), between them to create a multi-columned re-arrangeable browser.

While in the past these had to be made explicit by the system engineer, the algorithm has been extended to enable the automatic deployment of the interface for arbitrary RDF fragments, eliminating the need for engineered visualisation models. As such users can dynamically explore their personal information space, without requiring any effort on their part.

3.3 Multimedia Knowledge Management

The contextual log captured by the Semantic Logger has been presented as a means to annotate one's interactions. Part of the work undertaken in the PhotoCopain [31] project was to use content-based techniques to supplement the context-based information captured by the Semantic Logger, in order to annotate a user's photos. A set of low-level and mid-level feature extraction modules were created and the feature vectors created were used in reasoning tools and

[21]SPARQL W3C *http://www.w3.org/TR/rdf-sparql-query/*

classifiers to provide high-level semantic concepts for annotating the images in RDF. The software tools are now being incorporated into the OpenKnowledge system.

The AKTive Media [1] annotation user interface is also being adapted for use in the OpenKnowledge system. This tool supports the manual annotation of images for users who wish to further describe items in their collections. This can also be integrated into the OpenKnowledge system to visualise multimedia data using the visualiser interface defined in deliverable 5.1.

The available image-analysis modules from PhotoCopain that have been compiled into OpenKnowledge components are:

- **Scene Type Detector**: Detects whether the scene contains natural features of man-made features by using an edge-direction coherence vector. This assumes that man-made objects generally have strong straight-lines in them.

- **Face Detector**: Detects faces in the image using colour coherence providing their location, size, and count.

- **EXIF Extractor**: Extracts the EXIF data from an image providing many useful values for reasoning, such as the time and date of the image capture, details about the camera and capture conditions, such as whether the flash went off and whether the photo was taken with a wide-angle or telephoto lens.

Further reasoning could take place over the results from these modules, e.g. to decide if the photo is a portrait of a person (by aggregation of the camera details and the face detector results).

The annotations these media analyzers generate can be appended to a description of the media that is already existing in MPEG-7 format, as long as the OKCs that implement these analyzers understand MPEG-7. To that end, we have created a class from which image analysis OKCs that need to invoke MPEG-7 functions can extend, as well as a library they can include directly. The library provides methods for encoding and decoding MPEG-7 descriptors, as well as adding new descriptors into a current encoding.

3.4 MPEG-7 and Translators

When a user uploads images to their PhotoCopain account, it then creates meta-data descriptions of their images. As OpenKnowledge does not assume any particular data formats, these meta-data descriptions can be of any type. However, we show that using a standardised format, MPEG-7 in RDF, allows for other components to re-use the data with ease. The MPEG-7 translation is performed by, or between, the image annotator and the uploading of the information to the user's knowledge-base.

A component on the system that is used for translation of MPEG-7 concepts allows the interaction model to access the values of the MPEG-7 metadata and hence allows non-MPEG-7 compatible components to make use of the data.

The user installs on their machine a script which is responsible for gathering photographic information from the user's machine, annotating that data (see section 3.1.2), and then placing the annotations into the user's personal knowledge base, which they may or may not share with the community.

From this aggregated data, identifying events that have occurred at the same place and time, annotations can be shared between heterogeneous objects. For example, if a number of faces are detected in an image that was taken at a time where a meeting with named people is scheduled in a calendar item, these names could be used as suggested annotations for the picture.

3.5 Visualisation and Community Building

Finally, we present a mechanism for generating narratives from a user's annotated multimedia collection. Photographs are annotated and the annotations stored in the user's knowledgebase along with the rest of the user's captured metadata. The contextual information is then utilised to aid the generation of controlled narratives that provide an overview of a given event.

Model 3 shows an interaction model for the user–centered narrative generation process. This interaction results in both a multimedia presentation and an interaction model that contains the choices made by the user. By writing another interaction model as output from the narrative generation, these narratives can be shared thereby supporting community building through the publishing of generated narratives. These generated models do not necessarily require user interaction and would provide a means for 'replaying' the multimedia presentation, or for producing a new multimedia presentation based on new data. Model 4 shows an example of how this model may look for a particular instance.

3.5.1 Narrative Generation as a form of Visualisation

The generation of the narrative is an interactive process, requiring manual intervention by the user, alongside the automatic data gathering presented above. We use a generic interaction model for the narrative authoring process that is shown in Model 3. When a user initiates the narrative generation process, and an appropriate knowledgebase has been selected, the user is presented with possible starting points for the narrative based on classes and instances available in the knowledgebase. For example, if the user went on holiday to London, they may have uploaded all the photos they took on that holiday to their computer which their squirrel annotated and stored in their knowledgebase. This would allow the narrative generation algorithm to be seeded with the instance "London" of the class "place" over a given date range.

$\mathbf{r}(\text{user}, \mathit{initial})$
$\mathbf{r}(\text{narrativeGenerator}, \mathit{necessary}, 1)$

$\mathbf{a}(\text{user}, UID) ::$
  $\mathbf{a}(\text{user(SC,SI)}, UID)$ *then*
    $\mathit{results}(\mathit{Result}, \mathit{LCC}) \Leftarrow \mathbf{a}(\text{narrativeGenerator}, NID)$
      $\leftarrow \mathbf{showResults}(\mathit{Result}, \mathit{LCC})$

$\mathbf{visual}(\mathbf{showResults}(\mathit{Result}, \mathit{LCC}), \mathbf{narrativeGeneratorResults}(\mathit{Result}, \mathit{LCC}))$

$\mathbf{a}(\text{user(SC,SI)}, UID) ::$
  $\mathbf{null} \leftarrow \mathit{display}(C, I, UC, UI, \mathit{Quit})$ *then*
  $\mathbf{null} \leftarrow SC[UC|SC]$ *then*
  $\mathbf{null} \leftarrow SI[UI|SI]$ *then*

  $\mathit{searchCriteria}(SC, SI) \Rightarrow \mathbf{a}(\text{narrativeGenerator}, NID) \leftarrow \mathbf{not}(\mathit{Quit})$
  or
  $\mathbf{a}(\text{user(SC,SI)}, UID)$

$\mathbf{visual}(\mathbf{display}(C, U, UC, UI, \mathit{Quit}), \mathbf{narrativeGeneratorUI}(C, I, UC, UI, \mathit{Quit}))$

$\mathbf{a}(\text{narrativeGenerator}, NID) ::$
  (
    $\mathit{getClassesRequest}(R) \Leftarrow \mathbf{a}(\text{user}, UID) \leftarrow \mathbf{getClasses}(C)$ *then*
    $\mathit{getClassesResponse}(C) \Rightarrow \mathbf{a}(\text{user}, UID)$ *then*

    $\mathit{getInstancesRequest}(C) \Leftarrow \mathbf{a}(\text{user}, UID) \leftarrow \mathbf{getInstances}(C)$ *then*
    $\mathit{getInstancesResponse}(I) \Rightarrow \mathbf{a}(\text{user}, UID)$ *then*

    $\mathbf{a}(\text{narrativeGenerator}, NID)$
)
  or
  (
    $\mathit{searchCriteria}(SC, SI) \Leftarrow \mathbf{a}(\text{user}, UID)$
      $\leftarrow \mathbf{doSearch}(SC, SI, \mathit{Result})$ *then*
    $\mathit{results}(\mathit{Result}, \mathit{LCC}) \Rightarrow \mathbf{a}(\text{user}, UID)$
      $\leftarrow \mathbf{getLCC}(\mathit{Results})$ *then*

)

(3)

3.5.2 Narrative Generation as a form of Community Building

The output of the narrative generation is a multimedia presentation assembled from the knowledge stored about the user, retrieved from their knowledgebase based on the concepts they selected during the authoring process. As well as the visual output, the choices are encoded into another interaction model (such as that shown in Model 4) that can then be shared over the OpenKnowledge network. Conceptually this model is the *signature* of the previously generated narrative that can be re-instantiated using metadata of the same type from any knowledgebase. What this means is that executing a generated narrative interaction model is a more automated process that attempts to apply the narrative structure to new objects from a different knowledgebase. If it fails, due to unfulfilled requirements, similar classes can be identified in the user's knowledgebase, or alternatively, the executing peer could then use the OpenKnowledge platform to search for multimedia objects and execute an appropriate presentation.

$\mathbf{r}(\text{user}, \textit{initial})$
$\mathbf{r}(\text{narrativeGenerator}, \textit{necessary}, 1)$

$\mathbf{a}(\text{user}, UID) ::$
// **These are the choices our user has made**
$\mathbf{null} \leftarrow C = [\text{``}Place\text{''}, \text{``}Meeting\text{''}, \text{``}Face\text{''}, \text{``}DateRange\text{''}]\ and$
$I = [\text{``}MiltonKeynes\text{''}, \text{``}OpenKnowledge\text{''},$
$\text{``''}, \text{``}20070707 - 20070709\text{''}]\ then$

$chooseSimilar(C, I) \Rightarrow \mathbf{a}(\text{narrativeGenerator}, NID)\ then$
$getSimilar(SI) \Leftarrow \mathbf{a}(\text{narrativeGenerator}, NID)$
$\leftarrow \mathbf{checkSimilar}(C, I, SI)\ then$

$searchCriteria(C, SI) \Rightarrow \mathbf{a}(\text{narrativeGenerator}, NID) \leftarrow \mathbf{not}(Quit)$
$results(Result) \Leftarrow \mathbf{a}(\text{narrativeGenerator}, NID)$
$\leftarrow \mathbf{showResults}(Result)$

$\mathbf{visual}(\mathbf{checkSimilar}(C, I, SI), \mathbf{narrativeGeneratorCheck}(C, I, SI))$
$\mathbf{visual}(\mathbf{showResults}(Result), \mathbf{narrativeGeneratorResults}(Result))$

$\mathbf{a}(\text{narrativeGenerator}, NID) ::$
$chooseSimilar(C, I) \Leftarrow \mathbf{a}(\text{user}, UID)$
$\leftarrow \mathbf{getSimilarInstances}(C, I, SI)\ then$
$similarInstances(SI) \Rightarrow \mathbf{a}(\text{user}, UID)\ then$
$searchCriteria(SC, SI) \Leftarrow \mathbf{a}(\text{user}, UID)$
$\leftarrow \mathbf{doSearch}(SC, SI, Result)\ then$
$results(Result) \Rightarrow \mathbf{a}(\text{user}, UID)$

(4)

4 Conclusions

The aspiration of OpenKnowledge is to allow knowledge to be shared freely and reliably, regardless of the source or the consumer, but providing reliable semantic annotation of non-textual media is a non-trivial task. In this paper we have discussed one method for providing such interoperability by utilizing the MPEG-7 standard and its translations into OWL. Such a standard provides a base on which image analysis modules can build annotations, and, perhaps more importantly for OpenKnowledge, a base on which the ontology mapping techniques can be invoked.

To show how the system will work, we have presented a demonstrator that utilises OpenKnowledge for gathering of semantic data from a user's computer that involves annotation of images. The demonstrator gives an example of how this data might then be used to carry out interactive, iterative processes while recording the path of actions as automatically generated interaction models. The sharing of such automated versions of highly complicated protocols can in turn be used to support emergent communities of peers without the required facilities to carry out the initial interaction.

In addition, performing the software engineering exercise of porting an existing fully-fledged system to the OK architectures provides opportunities to assess and develop complex visualisation IMs to facilitate the interactions between user and knowledge base. This has resulted in implementing specialised visual terms for displaying forms and propagating input, as well as displaying images and RDF graphs.

References

[1] Fabio Ciravegna Ajay Chakravarthy, Vitaveska Lanfranchi, *Requirements for multimedia document enrichment*, 2006.

[2] H Alani, S. Dasmahapatra, K. O'Hara, and N. R. Shadbolt, *Identifying communities of practice through ontology network analysis.*, IEEE Intelligent Systems **18** (2003), no. 2, 18–25.

[3] T. Berners-Lee, *Keynote speech: Semantic web status and direction*, International Semantic Web Conference 2003(IWSC-03), Springer, Berlin, Heidelberg, 2003.

[4] T. Berners-Lee, J. Hendler, and O. Lassila, *The Semantic Web*, Scientific American **284** (2001), no. 5.

[5] J. Broekstra, A. Kampman, and F. van Harmelen, *Sesame: A generic architecture for storing and querying rdf and rdf schema*, 2002.

[6] Stavros Christodoulakis Chrisa Tsinaraki, Panagiotis Polydoros, *Integration of owl ontologies in mpeg-7 and tv-anytime compliant semantic indexing*, pp. 398–413.

[7] F. Dawson and D. Stenerson, *Internet calendaring and scheduling core object specification (icalendar)*, 1998.

[8] P. Duygulu, Kobus Barnard, J. F. G. de Freitas, and David A. Forsyth, *Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary*, ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part IV (London, UK), Springer-Verlag, 2002, pp. 97–112.

[9] S. L. Feng, R. Manmatha, and Victor Lavrenko, *Multiple bernoulli relevance models for image and video annotation.*, CVPR (2), 2004, pp. 1002–1009.

[10] R Garca and O. Celma, *Semantic integration and retrieval of multimedia metadata*, 2006, pp. 69–80.

[11] Jonathon S Hare and Paul H Lewis, *Saliency-based models of image content and their application to auto-annotation by semantic propagation*, Proceedings of the Second European Semantic Web Conference (ESWC2005) (Heraklion, Crete), May 2005.

[12] Jonathon S. Hare, Paul H. Lewis, Peter G. B. Enser, and Christine J. Sandom, *Mind the gap*, Multimedia Content Analysis, Management, and Retrieval 2006 (San Jose, California, USA) (Edward Y. Chang, Alan Hanjalic, and Nicu Sebe, eds.), vol. 6073, SPIE, January 2006, pp. 607309–1–607309–12.

[13] Stephen Harris, *SPARQL query processing with conventional relational database systems.*, WISE Workshops, 2005, pp. 235–244.

[14] J. Hunter, *Adding multimedia to the semantic web - building an mpeg-7 ontology*, 2001.

[15] J. Jeon, V. Lavrenko, and R. Manmatha, *Automatic image annotation and retrieval using cross-media relevance models*, SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval (New York, NY, USA), ACM Press, 2003, pp. 119–126.

[16] Jiwoon Jeon and R. Manmatha, *Using maximum entropy for automatic image annotation.*, CIVR (Peter G. B. Enser, Yiannis Kompatsiaris, Noel E. O'Connor, Alan F. Smeaton, and Arnold W. M. Smeulders, eds.), Lecture Notes in Computer Science, vol. 3115, Springer, 2004, pp. 24–32.

[17] Victor Lavrenko, R. Manmatha, and Jiwoon Jeon, *A model for learning the semantics of pictures*, Advances in Neural Information Processing Systems 16 (Sebastian Thrun, Lawrence Saul, and Bernhard Schölkopf, eds.), MIT Press, Cambridge, MA, 2004.

[18] Ryan Lee, *Scalability report on triple store applications*, Tech. report, MIT, 2004.

[19] A. Loizou and S. Dasmahapatra, *Recommender systems and ontology alignment in the semantic web*, ECAI2006 Workshop on Recommender Systems (Riva del Garda, Italy), 2006.

[20] Antonis Loizou, *Unlocking the potential of recommender systems: Achieving multiple domain recommendations. mini-thesis: Phd upgrade report.*, Tech. report, Electronics and Computer Science, University of Southampton., 2007 (In press).

[21] m. c. schraefel, M. Karam, and S. Zhao, *mSpace: interaction design for user-determined, adaptable domain exploration in hypermedia.*, Proceedings of AH 2003: Workshop on Adaptive Hypermedia and Adaptive Web Based Systems. (Nottingham, UK) (P. De Bra, ed.), Springer, 2003, pp. 217–235.

[22] B. McBride, *Jena: Implementing the RDF model and syntax specification.*, Proceedings of the 2nd International Workshop on the Semantic Web, SemWeb, 2001.

[23] Florent Monay and Daniel Gatica-Perez, *On image auto-annotation with latent space models*, MULTIMEDIA '03: Proceedings of the eleventh ACM international conference on Multimedia, ACM Press, 2003, pp. 275–278.

[24] Y. Mori, H. Takahashi, and R. Oka, *Image-to-word transformation based on dividing and vector quantizing images with words*, Proceedings of the First International Workshop on Multimedia Intelligent Storage and Retrieval Management (MISRM'99), 1999.

[25] David Robertson, *Multi-agent coordination as distributed logic programming.*, ICLP, 2004, pp. 416–430.

[26] Kilian Stoffel, Merwyn G. Taylor, and James A. Hendler, *Efficient management of very large ontologies*, AAAI/IAAI, 1997, pp. 442–447.

[27] Michael Streatfield, *Report on Summer Internship Work For the AKT Project: Benchmarking RDF Triplestores*, Tech. report, University of Southampton, 2005.

[28] J. Tang, J. S. Hare, and P. H. Lewis, *Image auto-annotation using a statistical model with salient regions (speech)*, Proceedings of IEEE International Conference on Multimedia & Expo, 2006.

[29] Raphal Troncy and Jean Carrive, *A reduced yet extensible audio-visual description language: How to escape from the MPEG-7 bottleneck*, October 2004, pp. 87–89.

[30] Mischa M. Tuffield, *An investigation into automatically captured autobiographical metadata, and the support for autobiographical narrative generation. mini-thesis: Phd upgrade report.*, Tech. report, Electronics and Computer Science, University of Southampton., 2006.

[31] Mischa M. Tuffield, Stephen Harris, David P. Dupplaw, Ajay Chakravarthy, Christopher Brewster, Nicholas Gibbins, Kieron O'Hara, Fabio Ciravegna, Derek Sleeman, Nigel R. Shadbolt, , and Yorick Wilks, *Image annotation with photocopain*, Proceedings of the First International Workshop on Semantic Web Annotations for Multimedia (SWAMM) (Edinburgh), May 2006, held as part of 15th World Wide Web Conference (22-26 May, 2006).

[32] Mischa M Tuffield, Antonis Loizou, David Dupplaw, Sri Dasmahapatra, Paul H Lewis, David E Millard, and Nigel R Shadbolt, *Semantic logger: Supporting service building from personal context*, Proceedings of Capture, Archival and Retrieval of Personal Experiences (CARPE) Workshop at ACM MM, ACM MultiMedia, October 2006.

[33] World Wide Web Consortium, *SPARQL query language for RDF, working draft*, Tech. report, World Wide Web Consortium, 2005.