Deliverable D.4.2
Executive Summary

Knowledge Lenses and Process Support Tools

Authors: Victoria Uren, Open University, v.s.uren@open.ac.uk

Sam Chapman, University of Sheffield, sam@dcs.shef.ac.uk

Aba-Sah Dadzie, University of Sheffield, a.dadzie@dcs.shef.ac.uk

Piercarlo, Slavazza, Quinary S.p.a., x-media@quinary.com

Thomas Franz, University of Koblenz-Landau, franz@uni-koblenz.de

Yuangui Lei, Open University, y.lei@open.ac.uk

: Johannes Busse, Ontoprise, busse@ontoprise.de

Daniel Rabus, Ontoprise, rabus@ontoprise.de

Roman Korf, Ontoprise, korf@ontoprise.de

Tran duc Thanh, University of Karlsruhe, dtr@uni-karlsruhe.de

Till Christopher Lech, CognIT, Till.Christopher.Lech@cognit.no

Work-package: WP4 Knowledge sharing and process support

Type: Technical Report

Distribution: Public

Status: Preliminary

Date: 21.12.2007

Deliverable Coordinator: Victoria Uren, Open University

Reviewers: Steve Fullerton/Andrew Maisey, Solcara, Christine Preisach, University of Hildesheim

Area Coordinator: Steffen Staab, Univerity of Koblenz-Landau

Project Coordinator: Fabio Ciravegna, University of Sheffield

EU Project Officer: Ralph Feirgolla

TABLE OF CONTENT

1 Introduction

The X-Media consortium has responded to the requirements for knowledge sharing and process support with a range of task centric tools that are individually innovative while working within the constraints of the X-Media architecture to facilitate interoperability. This report presents technical descriptions of the knowledge sharing and process support prototype tools which exploit the semantic metadata extracted by Area 2 partners and which operate with the X-Media kernel.. It describes tools for annotation, semantic search and browsing, presentation of results and analysis and sharing of results.

2 The Tools

In this section we present a very brief description of each tool.

2.1 The Open Document Format (ODF) Annotation Toolbox

Annotating text or images is narrowly related to three typical other tasks: identifying the media which are to be annotated interactively, triggering IE and text annotation engines of area2 in order to get pre-annotated media, and calling the interactive annotating component possibly by telling it which image or text there has to be annotated. Three flavours of office documents annotation tool allow for annotating regions of Open Document Format (ODF) text, presentation and spread sheet documents, these are: Text (ODT) for text fragments, Presentation (ODP) for slides, and Spreadsheets (ODS) for ranges of cells.

2.2 Image Annotation

Beside the ODF annotation components an image annotation component which allows for annotating and amending annotations of regions within images was developed. This component differs from the former components in that it can be run as a java stand alone application (see Figure 1).

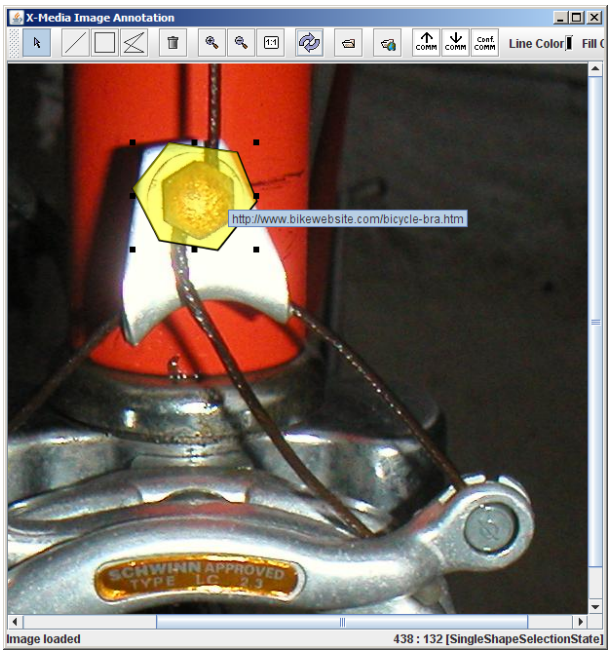

Figure 1 Here the user navigated to a page about bike brake adjustment. She highlights the URL, grabs it with the mouse and drops it to the region previously selected in the annotation tool.

2.3 LENA

LENA[1] stands for **LE**ns based **NA**vigator. LENA supports viewing RDF data in a web browser, rendered according to the lens descriptions provided. The use of multiple lenses is supported and these are indicated when available for a resource. The design goal is to make different views onto the same data easily accessible for the user. As shown in Figure 2. Knowledge lenses are one of the contributions of the X-Media project planned for WP4. LENA implements one interpretation of a knowledge lens based on an interpretation of RDF data towards different visual representations. Such different views are required in several of the X-Media use cases

Figure 2: Screenshot of LENA. LENA lists available lenses in a box (named *Default Lenses*) on the left. The box below (named *Included Classes*) lists RDF classes referenced in the underlying RDF repository and the number of instances for each class. Furthermore, a snowball-like icon indicates that lenses are defined at least for some of the instances.

1 http://isweb.uni-koblenz.de/Research/lena

2.4 SemSearch

SemSearch comprises i) a query interface, which supports the specification of multi-keyword queries; ii) a keyword search engine, which makes sense of user queries by exploiting the domain ontology and the extracted metadata; iii) a query translation engine, which derives appropriate formal queries from the user query; iv) a query refinement engine, which allows the user to reformulate their query towards their information seeking needs. The system also provides an index engine, which indexes semantic entities contained in the domain ontology and the gathered metadata repositories.

One key contribution of the current SemSearch prototype is its mechanism for keyword based querying of semantic metadata, which makes semantic search intuitive for ordinary end users. Particularly, it allows users to make searches without having to know either the structure of the ontology, tackling the issue of *understanding* the semantic space, or a formal search language.

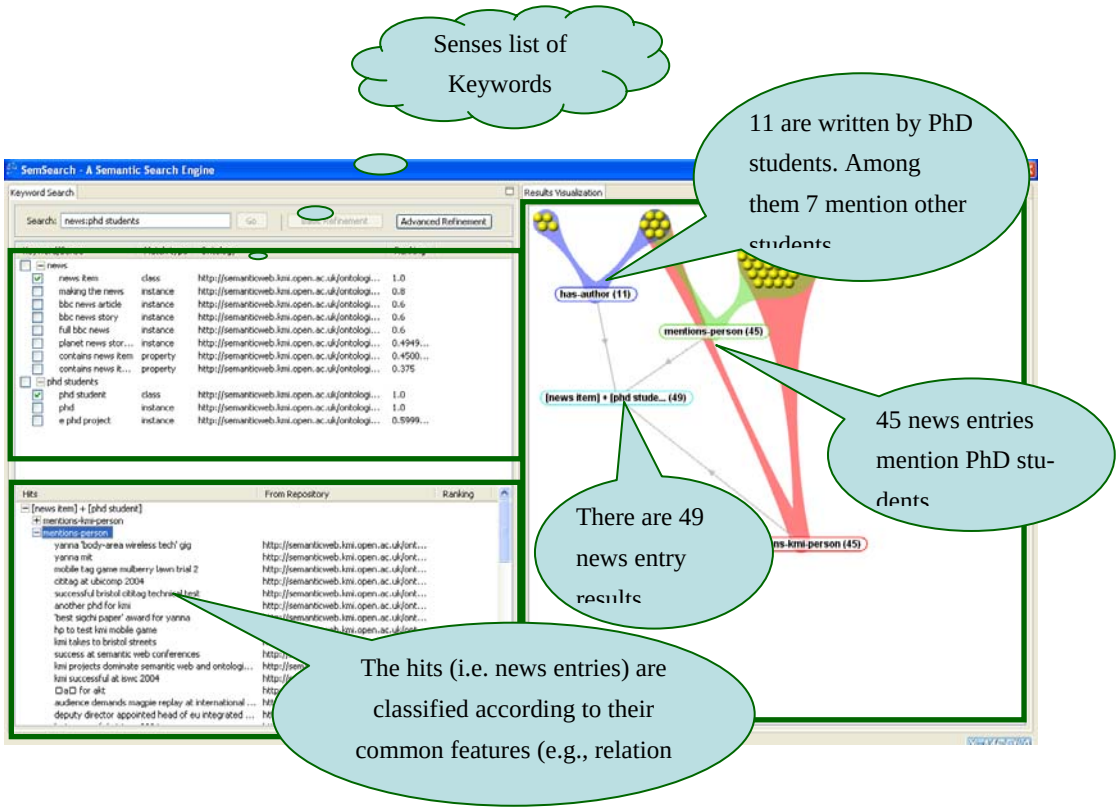

Figure 3 Screenshot of SemSearch showing the user viewing results from a basic search

2.5 K-Search

K-Search is a semantic search interface capable of bringing together the worlds of semantic conceptual search (structured knowledge) and also keyword search (unstructured knowledge). These search modalities can be utilised independently as well as together in a hybrid modality. These novel interaction methods allow flexible searching of knowledge which may only be partially represented in the domain ontology.

Structured and unstructured data both have advantages and disadvantages; one providing order and precision, the other flexibility of use. To take advantage of both, K-Search uses a dual store (K-Store): a normal keyword index is used alongside a semantic index storing structured triples.

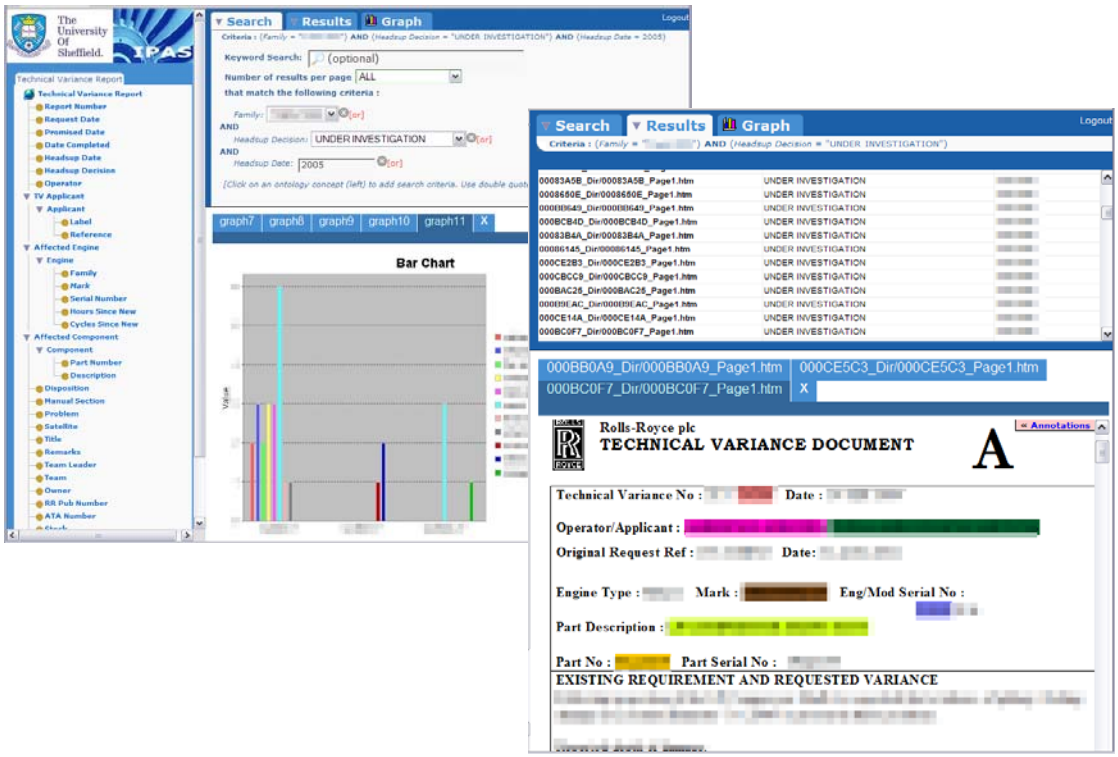

Fig. 4. K-Search querying interface and visualisation of results. When a query is performed, the result set contains the reports where the concepts and the keywords in the query co-occur. The set is displayed as a list in the mid-right panel of the interface; each item in the list shows the name of the document and the values of the fields used for OS. Individual reports are displayed on the bottom right on request (by clicking on the file name for a list item). Multiple documents may be opened simultaneously, each displayed in a different tab. The original layouts of the documents are maintained. Please note that portions of the text have been deliberately obscured for reasons of confidentiality.

2.6 XXploreKnow!

XXploreKnow! is an ontology-based application, which supports the exploration and retrieval of documents and facts contained in knowledge bases. It has four distinctive features: 1) it is capable of translating an unrestricted number of keywords to a ranked list of interpretations, 2) it combines factual search with document retrieval, 3) the implemented data filtering mechanism allows the user to create customized views on the KB, 4) XXploreKnow! features personalization functionalities, which are based on a rich ontology-based model of the context.

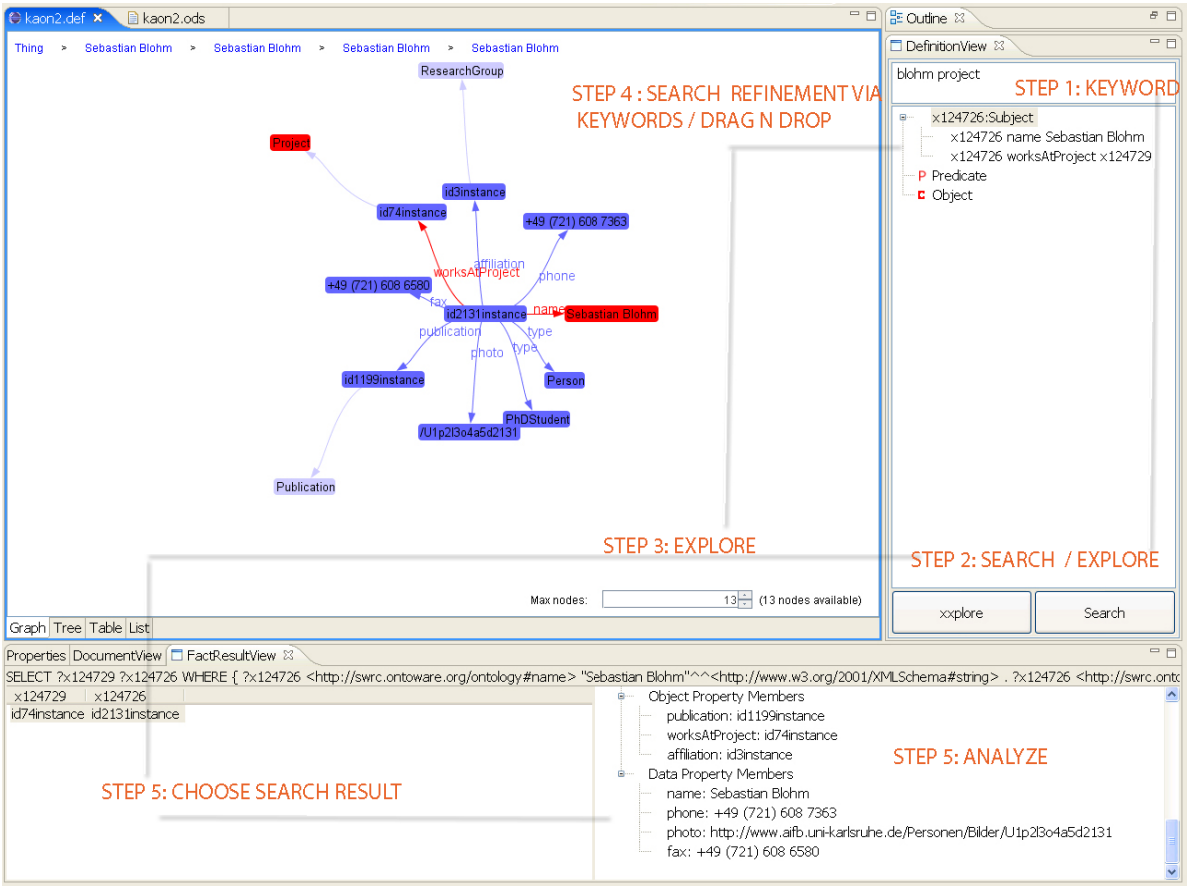

Fig. 5 XXploreKnow! (STEP 1), the user enters keywords which are interpreted to produce a list of possible queries. The user selects the intended query and the system visualizes the corresponding portion of the schema. The user may either activate the “search" or the “xxplore" button (STEP 2). With “xxplore" (STEP 3a), the user can expand nodes shown in the visualization of the query to traverse to neighbouring elements. During this exploration, the user can drag and drop elements from the schema view and the concept outline view to the query definition view (STEP 4) to further refine the query. With “search" (STEP 3b), the SPARQL query is sent to the query engine. The results, which may contain also inferred facts, are shown to the user (STEP 5).

2.7 K-Views

Good results presentation concerns the provision of intuitive filters that present knowledge retrieved from different perspectives, providing context that enriches the user experience. The evidence collection and analysis stages in the vision demonstrator identified two main methods for clustering/categorising data: the visual representation of classes derived from the causes ontology, and a *causes* trees to cluster evidence on each node of the tree.

The current presentation tool is built using the *prefuse* visualisation toolkit which provides support for data input using readers for GraphML (an XML format), *prefuse* data tables built from csv files, for instance, and from databases such as MySQL, with inbuilt support for querying using SQL. Figure 6 shows a network graph drawn using *prefuse*.

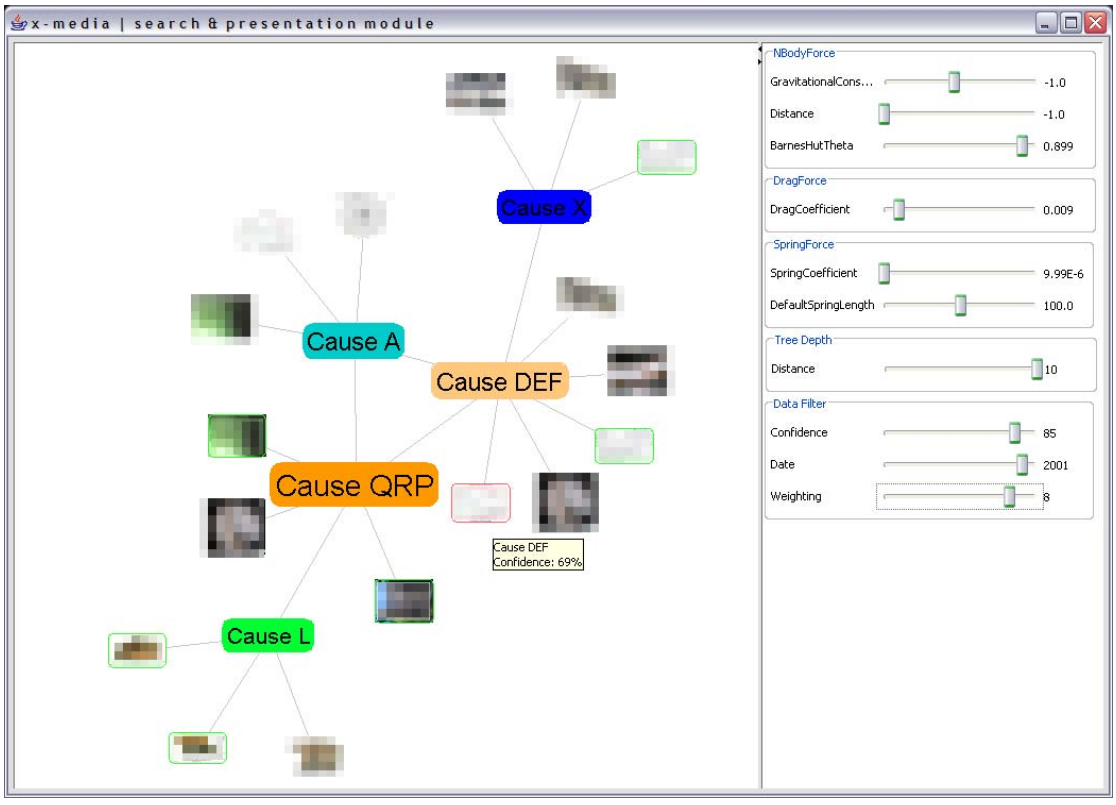

Fig. 12: A network graph drawn using the *prefuse* visualisation library, showing the relationships between concepts in a causes ontology, and thumbprints providing a preview of the assertions retrieved that contribute to the hypothesis for each concept being the root cause of the issue under investigation. Please note that portions of the image have been deliberately obscured for reasons of confidentiality.

2.8 The CORPORUM Summarizer

The main focus in X-Media will be the presentation of extracts of document on a given incident (i.e., issue resolution case) along a timeline, thus providing an overview of the development in the case, as illustrated in the mock-up below (Figure 7).. The main challenge in this context will be to identify new versus given information across multiple documents in order not to present identical text parts in different summaries that are presented to the user. The summarisation tool proposed X-Media is based on CognIT's CORPORUM Summarizer. This tool is an extraction-based summariser, developed within the OnToKnowledge project, building on CognIT's CORPORUM natural language processing tool OntoExtract. OntoExtract is an NLP tool, which analyses documents and extracts the core concepts and named entities as well as associations between them, thus establishing a lightweight concept graph. This semantic representation is then used as a basis for the choice of sentences to be included in the summary.

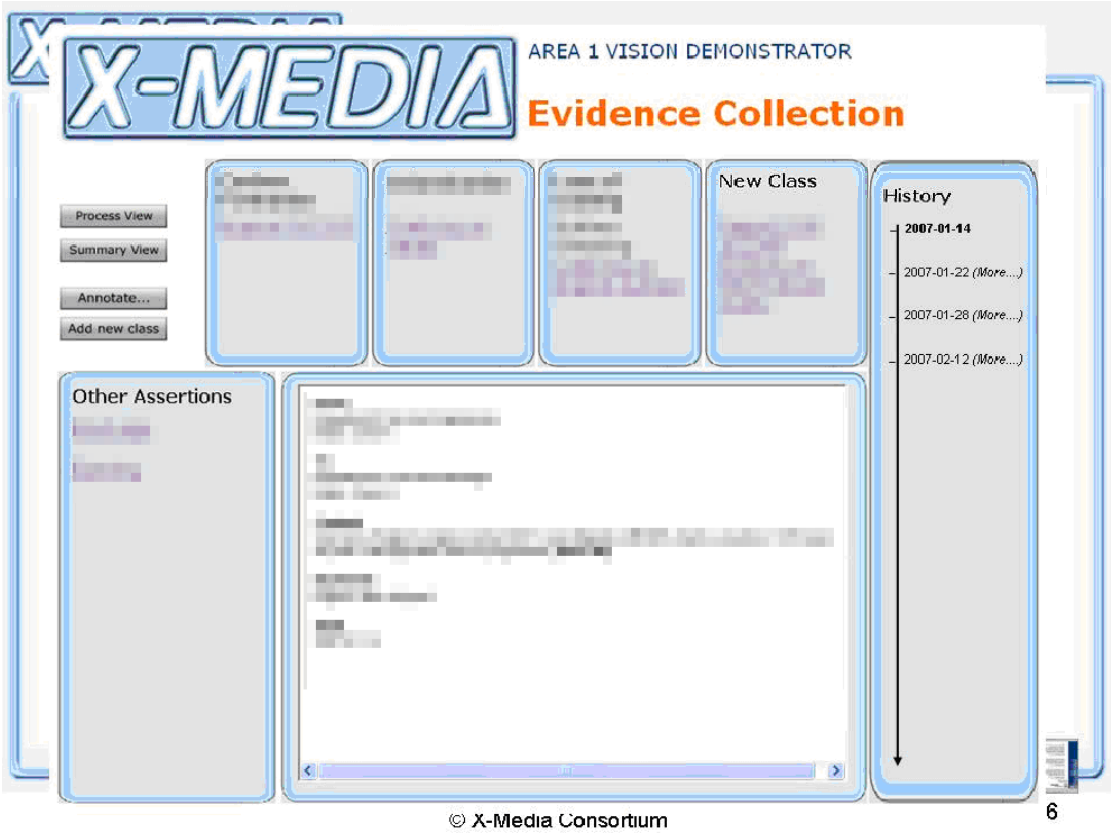

Figure 7. The summariser interface (vision demonstrator) . Please note that portions of the text have been deliberately obscured for reasons of confidentiality.

2.9 Semantic Scratchpad

The Semantic Scratchpad (or BOX) is a special personalized, task-based and virtual semantic repository storing task-specific facts. A single BOX object provides *read-only* access to a given subset of the knowledge stored within an X-Media system; means are provided to access such knowledge in tabular and graph format (however, it is not guaranteed that *both* formats are supported by a given BOX instance).

The Semantic Scratchpad is instantiated by means of a Java component named BOX-Handler (available through the X-Media Kernel), which manages BOX objects. In the first Scratchpad release, it is provided as a BOX implementation which selects knowledge by means of a SPARQL "SELECT" query dispatched to a given AccessKnow repository; such implementation provides access to the selected knowledge only in tabular format. At later stages of the project, other implementations may be provided which will enable access to the knowledge in graph format also, and to select knowledge by accessing the Kernel component XMSearchAndIndexHandler (see [D11.1]); moreover, more powerful ways of selecting knowledge will be also evaluated, such as Networked Graphs (see [Schenk 2007]).

As a sample usage of the BOX APIs, a (generic) BOX Browser web application will be provided, which will allow an X-Media user to login, manage his/her BOXes, and browse their content (possibly by exploiting the presentation facility of the X-Media knowledge lenses).

The BOX Browser code will provide a baseline that may be further customized by developers in order to build domain specific applications.

2.10 The Koblenz Email tool

Email is a vital tool for debate, knowledge sharing and distribution (e.g. by sending around attachments). One goal of the Koblenz Email tool is to provide emails and associated metadata about persons, tasks, and attachments to the X-Media knowledge base. Additionally, easy-to-use task management features will be supported focussing on the provision of task-awareness to users and leveraging retrieval based on metadata not only about documents, but also people and tasks. Moreover, services such as automatic notification based on different criteria will be supported, e.g. notification based on a newly defined issue resolution process. The tool is implemented as a plugin for the Thunderbird email client. It makes use of the X-COSIM framework that is partly used in X-Media knowledge representation for storing email, person, and task metadata compliant to the X-Media KB.

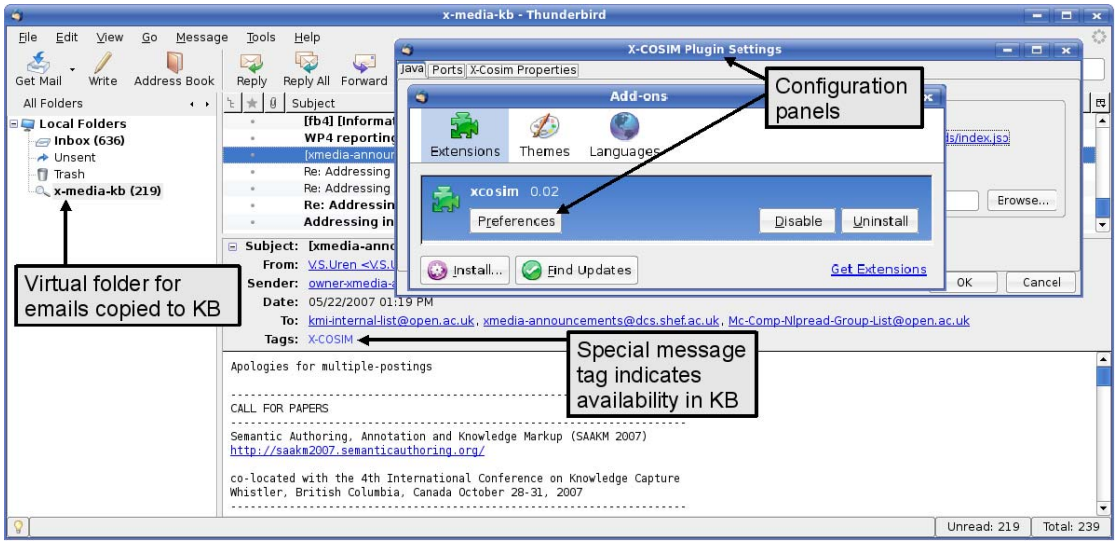

Fig 15 The semantic email client

3 Conclusions

The deliverable reported our progress in developing the knowledge sharing tools that will be used to build the first phase of X-Media test beds. In summary our achievements have been:

- Building annotation tools for open document format and images which can annotate regions.
- Four different search and browse tools each exploring different research lines on user interaction and search refinement. Search is a central knowledge sharing task and the comparison of multiple search modes will allow us to evaluate the advantages and disadvantages of different approaches.
- Innovative work has been undertaken on the presentation of search results, particularly of summary views of semantic data.
- A ground breaking semantic scratchpad prototype has been built. The scratchpad idea came directly from requirements work with users and is central to the envisioned X-Media knowledge sharing and reuse paradigm.
- A first prototype of a semantic email tool has been built to demonstrate key process support ideas.

Already in phase 1, these tools address many of the X-Media requirements. These are itemised for each tool but here we note particularly that the tools all address the issue of interoperability. In this phase, interoperability is gained through compliance with the X-Media kernel and through the wide use of the Eclipse RCP which will facilitate the packaging of task driven tools into use case driven platforms. In addition, user-centred design is important in many of the tools, whether through better search support or the presentation of knowledge, using technologies like Fresnel lenses or summarisation of text or semantic metadata.