Toward a Versatile InfoRmation Toolkit for end-Users oriented Open-Sources explOitation : VIRTUOSO

Géraud Canet*, Gael de Chalendar*, Laurent Dubost**, Stéphan Brunessaux***
Gérard Dupont***, Axel Dyevre****, Khaled Khelif***, Bruno Quere**

*CEA-LIST ; Laboratoire Vision et Ingénierie des Contenus ; 18, rue du Panorama ; BP 6
92265 Fontenay aux Roses Cedex ; France
geraud.canet@cea.fr
gael.de-chalendar@cea.fr
**Thales, avenue carnot, 91883 Massy Cedex
laurent.dubost@fr.thalesgroup.com
***Cassidian, IPCC, Parc d'Affaires des Portes - BP 613 - 27106 Val-de-Reuil Cedex
France
gerard.dupont@cassidian.com
khaled.khelif@cassidian.com
****CEIS ; 280, boulevard saint-germain, 75007, Paris
adyevre@ceis.eu

Abstract. This paper describes the FP7-Sec VIRTUOSO project. This project aims at providing technical framework for the integration of tools for collection, processing, analysis and communication of open source information.

1 Introduction : context and motivation

The explosion of on-line information (textual information as well as other unstructured sources such as audio, images, and video) motivates most current work in content analysis and knowledge extraction. Although even if massive volumes of information are available at low cost as free textual, audio or video contents, people cannot read and digest this available information as fast as it is published and they are still looking for tools to mine the acquired content and allowing them to concentrate their attention and effort only on the valuable information. Content mining is the generalisation of text mining that can be seen as "the discovery by computer of new, previously unknown information, from different written resources". The acquisition of knowledge from unstructured data gives us the need to combine a wide variety of technologies. It needs also to be able to fuse information extracted from heterogeneous resources. The exploitation of the acquired knowledge to support the decision-makers is another challenge that will be addressed. The goal of the VIRTUOSO project is to innovate in the following items:

- **End-users' involvement**: integrating the end-users in the conception at all steps of the development/evaluation process.
- **Interoperability**: developing a system for managing and processing open source information, by developing new specific components but also by reusing and integrating available off-the shelf methods and tools. Satisfying the interop-

erability constraint during the design and the realisation of the platform by developing methodology, tools and supporting standardisation to encounter obstacles to interoperability between the technologies is the key success of proposal system.

- **Information Extraction**: developing an increasing approach based on an Open Information Extraction starting from domain specific (supervised information extraction) to domain independent (unsupervised information extraction, e.g. machine-learning)
- **Knowledge building (VKB):** developing a standard acquisition of knowledge from information extraction
- **Decision Support**: developing tools based on the acquired VKB Virtuoso Knowledge Base, to assist the decision-makers in their activities
- **Computing**: studying and evaluating the computation power needed for given application the volume of data

2 Objectives of the VIRTUOSO framework

VIRTUOSO will provide a technical framework for the integration of tools for collection, processing, analysis and communication of open source information. "Plug and play" functionalities that improve the ability of border control, security and law enforcement professionals to use data from across the source / format spectrum in support of the decision making process will be enabled by this middleware framework. As a proof of concept and to highlight the efficiency of this open-source code framework, a prototype will be built and demonstrated using operational scenarios. The project, developed by a consortium of 17 European organisations, is co-funded under the FP7 programme of the EU and will comply with legal considerations, enforcing the principles of privacy and data protection to ensure the interests of citizens within the European Union.

The main functional components of the VIRTUOSO system are:

Information Gathering components: collects the information coming from multiple supports (disks, intranet, the Internet…), multiple format (paper, digital, analogical...), and of multiple natures (textual, audio and visual (image and video), and then converts data into standard format. For instance paper documents will be scanned and translated via OCR API into XML output format, in the same way the speech audio documents will be converted, using speech to text API, into XML format.

Information Extraction and Structuring components: Structuring Open Source Information to Support Intelligence Analysis. It deals with information reduction and structured based on event and information extraction. It comprises information filtering and categorization, Information Retrieval, Entity and event extraction, Information summarization.

Knowledge Acquisition components: These services deal will the storage and the representation of the knowledge built from the extracted information;

Decision support components: The VIRTUOSO Decision-maker support toolkit will be developed to provide a substantial aid to the decision makers by integrating various sources of information and by developing tools to exploit the knowledge stored in the VKB and components for Situation Assessment, Scenario Building and Misinformation detection. It will provide also an intelligent access to or visualize relevant knowledge, and aid the process of structuring the work and the decisions.

3 Overall architecture

The VIRTOSO framework will rely on a "Service Oriented Architecture" (SOA) as the core paradigm for the design and integration of components. Each component that could be integrated in the platform shall implement one or several functionalities that are described by service interfaces. The function workflow needed to provide user applications will be done by putting together services and calling them in the right order. Each component, implementing one or several service interfaces won't have any knowledge of the other services and their capabilities. They will provide to the others one or several processing capabilities (i.e. services) which will be driven by the orchestrator to define business processes. As a consequence, the service definition and conception is a key step in the platform. The granularity of the services should be one of the main concerns during the design and development of a WebLab component.

In order to provide a flexible architecture, the service design should respect the following features:

- Loosely-coupled: It means that services should be as autonomous as possible and that dependencies between components should be avoided;
- Coarse-grained: A service should provide a coherent set of functions and should hide implementation complexity;
- Standardized interfaces: A service should implement one or more standardized interfaces from the service taxonomy and reuse the generic service interfaces. It gives access to the functions provided by the service;
- Integrable: A service can be easily integrated in a global application and be used in a chain with other services.

Every service is provided by a "producer" to a "consumer". The interaction between producer and consumer is carried out by a middleware (the enterprise service bus) that is responsible of the mediation and the communication between the services. As a consequence, a consumer invokes a service to carry out a function of which he is the producer.

A directory of services is provided in the platform to allow the producers to publish their service offers and enable the chain builder to select the right service on each step of a processing orchestration. A service call is done by sending a message through the middleware from the consumer to the publisher either in an asynchronous or a synchronous way.

In the architecture, we will consider:

- Business services that provide business functions (such as video segmentation, text clustering);
- Technical services that are part of the provided baseline (such as security, data access layer, etc.);
- Graphical User Interface (GUI) components that will interact with users on one side and with the service bus on the other side to request process or data.

In Virtuoso, the various components are splitted into four different and mainly autonomous frameworks responsible of different kind of things. They are the Processing framework, the Decision Support framework, the Evaluation framework and the Unified Presentation framework (see Figure 1). Besides the frameworks, there is (possibly several) knowledge bases used to store all the data extracted from components and consolidated by others.

The Virtuoso applications need to physically separate some parts of the system, such as data acquisition and preliminary transformations on the one hand and data processing, storage and exploitation on the other hand. The transfer of data from the first part to the second one has to be specified too.

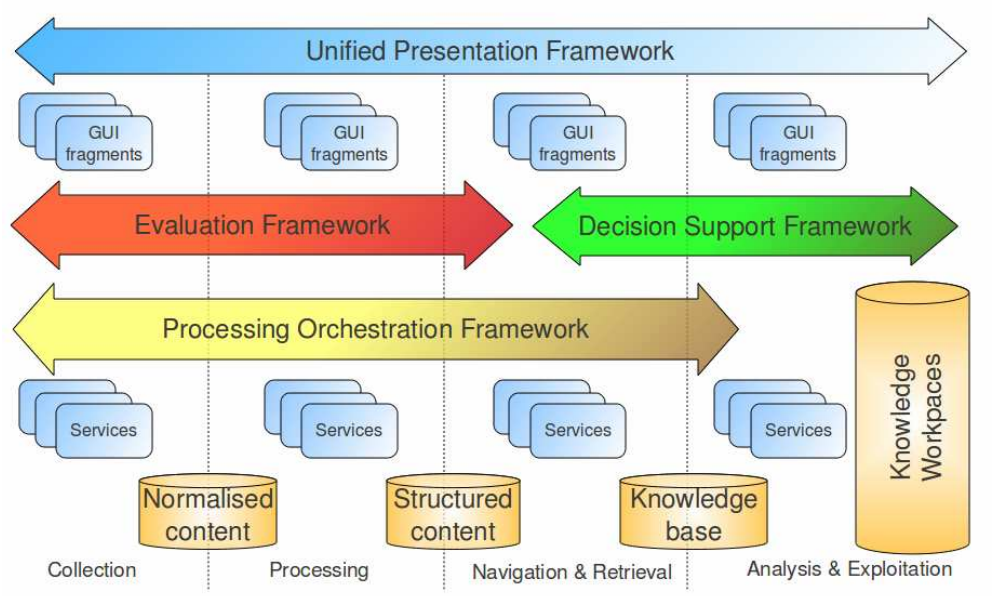

Figure 1 - The Virtuoso Architecture

3.1 The processing framework

All processing components developed in the context of the VIRTUOSO project will be integrated as services using the WebLab platform. The WebLab project is an open source framework developed and maintained by EADS (Giroux et al., 2008). Here, we summarize the WebLab project and its organization. This platform was initiated in the WebContent research project[1].

The WebLab platform provides a set of tools and methods that aim at building information systems for intelligence applications in the economy, strategy and military domains.

[1] http://www.webcontent.fr

Typically, WebLab handles multimedia and unstructured data (text, image, audio and video) through heterogeneous business components providing services. WebLab is made of 3 layers (Figure 2):

- the **WebLab Core** which represents the technical base designed to make heterogeneous components cohabit and work together within a service oriented architecture (SOA),
- the **WebLab Services** which are shown as a set of coherent software services dealing with elementary functionalities as well as GUI components which can be assembled to achieve custom-built applications, and
- the **WebLab Applications** which are the result of the integration of the WebLab Services based upon a common exchange model and common services interface provided by the WebLab Core. Thus, it guarantees a minimal effort for programmers to integrate their tools into a complete application.

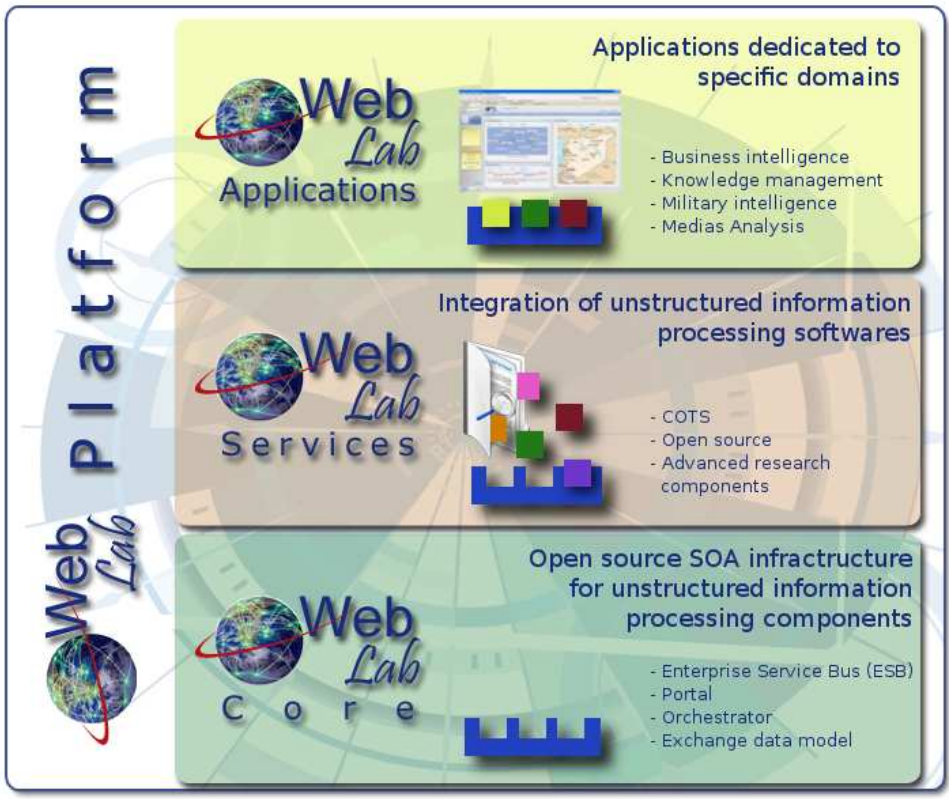

Figure 2 - The WebLab core, services and applications

WebLab relies on many standards, including semantic Web standards. In addition to these standards, WebLab proposes a data exchange model allowing heterogeneous components to communicate with each other (see Figure 3).

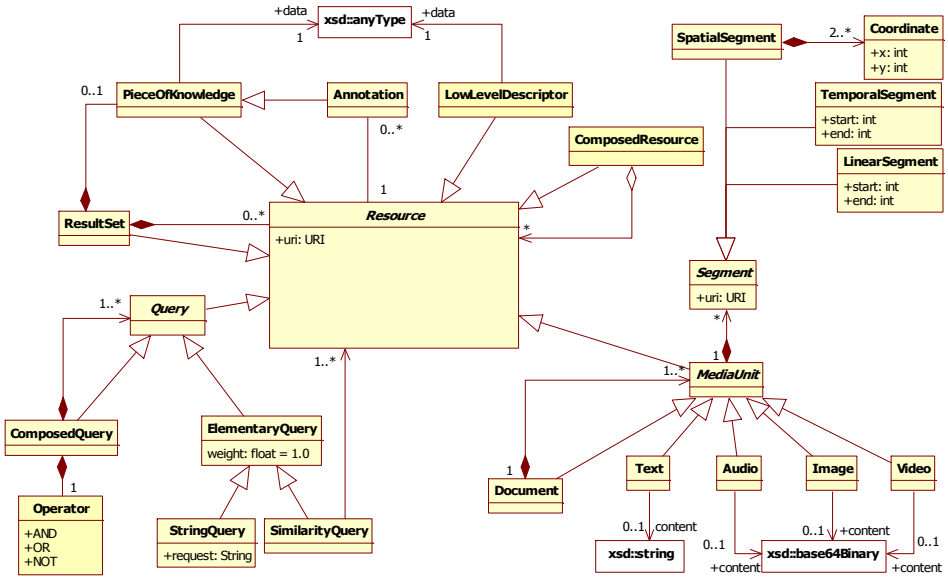

Figure 3- The WebLab exchange model

This exchange model is mainly organized around the *Resource* object which is a generalization of the different types of entities to manage by the platform. The *MediaUnit* object allows describing each information entity (audio, video, image, text) that will have to be managed by the different technical component. The model also enables to annotate the resources with RDF annotations (subject- predicate-object triples).

3.2 The decision support framework

In Virtuoso, Decisions Support and Visualization tools aim at exploiting the information produced by the processing and/or the users and stored in Knowledge Base.

These Decision Support and Visualization modules in VIRTUOSO will be integrated on top of a Decision Support Framework based on the ING Semantic Middleware. ING Semantic Middleware is developed and maintained by THALES. Its objective is to facilitate the development and integration of Decision Support and Visualization applications in the fields of Security, Defence, and Intelligence.

The semantic representation (i.e: by triples such as : Bruno “works for” Thales) may bring great operational benefits. It offers a great expressivity and its flexibility facilitates the interoperability of information systems. Moreover, the semantic representation is very well suited for the representation and computation of social networks which are crucial in security applications. In spite of all these potential operational benefits, semantic representation is rarely used in operational security systems, because operational knowledge is often complex and requires to be handled at a certain a level of abstraction (otherwise developers could easily be “overwhelmed by triples”!). ING solves this problem.

ING provides applications developers with high level standard interfaces to query, update a semantic knowledge base, compare and merge pieces of knowledge and enable adaptation (and hot deployment) of the knowledge base model.

Virtually, these standard interfaces enable application developers to use any database as a semantic repository, provided they use/develop the right connector. In VIRTUOSO, ING uses a connector to the ITM semantic repository developed by MONDECA.

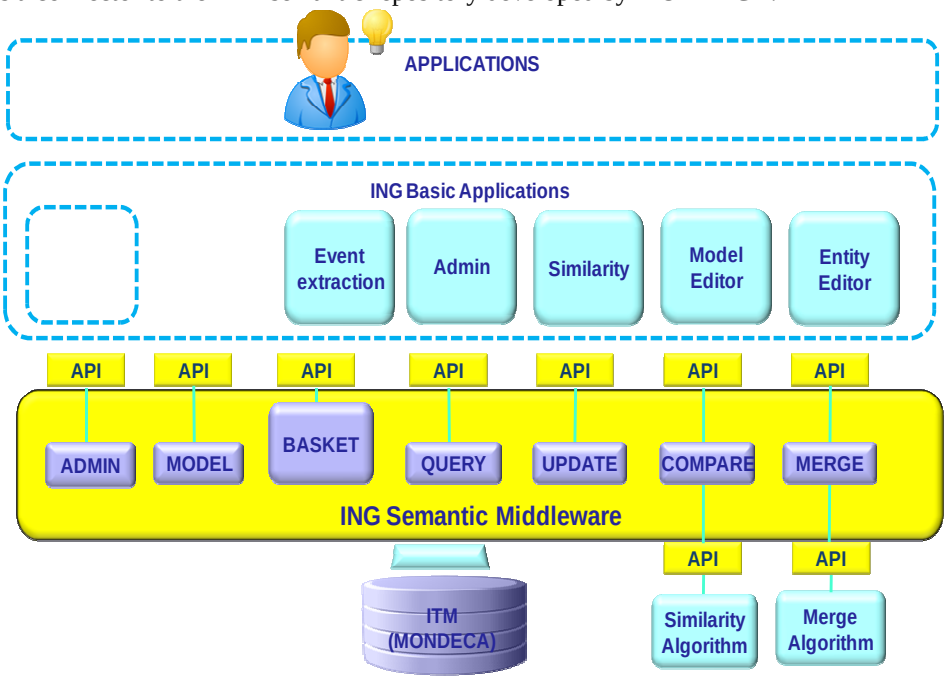

Figure 4 - ING Semantic Middleware APIs & basic applications

All applications developed on ING use the "Basket" service to communicate. Via the basket, the result of a query, can be displayed by another application, such as a graph viewer. Applications read their inputs in the basket and write their outputs in the basket. As a result, the basket provides a simple and practical journalizations of users operations, each operation being a view (i.e: layer) on the knowledge base.

In addition to these middleware features, Thales has developed several basic applications on ING entity edition, model edition, collective data fusion, event extraction… these basic applications are brought to the VIRTUOSO project.

The technical choices of ING Semantic Middleware (J2E, SOA, web client…) make the VIRTUOSO Decision Support Framework fully compatible with Processing Framework based on the WEBLAB project .

The release of the ING middleware as an open source software is currently being considered by Thales within the VIRTUOSO project.

3.3 The evaluation framework

The aim of the evaluation framework is to define a standardized business process to evaluate the scientific quality of the various tools issued from the processing framework. Evaluations will internally use datasets and evaluation software proposed by international evaluation campaigns, such as TREC for information retrieval for example. These tools and datasets will be wrapped inside Web services implementing standard WebLab interfaces. Figure 5 shows the envisioned architecture of an evaluation setup using this evaluation framework.

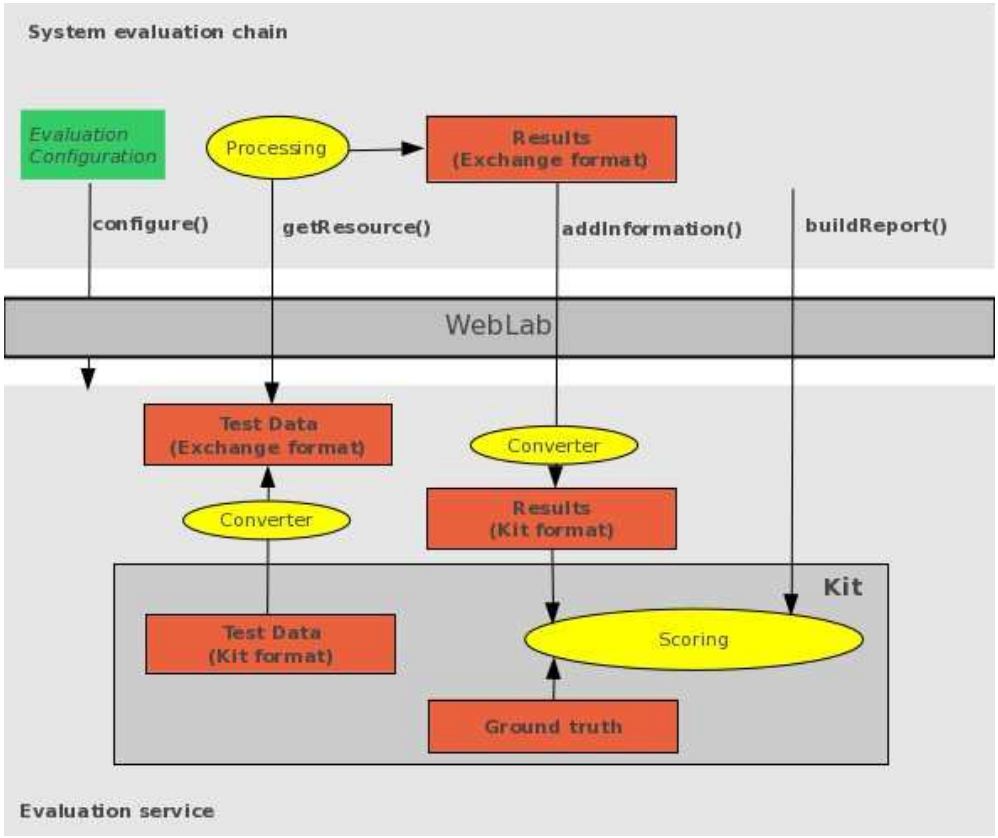

Figure 5 - The evaluation process

An evaluation service must implement three WebLab interfaces:

- *Configurable* to be able to set global information related to an evaluation, such as the run name;
- *SourceReader* to enable the service being evaluated to retrieve test data in WebLab data exchange format;
- *ReportProvider* to allow the service being evaluated to submit its data in WebLab data exchange format and to get the evaluation result as a WebLab document, results being expressed in XML/RDF.

Thus the creation of a Virtuoso evaluation service consists in writing programs able to:

- convert the test data format into the WebLab exchange model format;
- convert the WebLab exchange model format to the submission format used in the evaluation campaign;
- express the evaluation result as XML/RDF and generate a result report in the WebLab exchange model.

For each targeted evaluation campaign, it will be necessary to define an ontology enabling to express the evaluation data such as scores. We will try to use as much as possible a core evaluation ontology for standard elements such as precision and recall and specialized ontologies for the kind of data specific to a campaign.

The work on the evaluation framework has just started. Currently, the list of evaluation kits necessary to evaluate all the kind of processing components that will be developed during the Virtuoso project is not complete. Some of the kits considered are:

- Named Entities Extraction, using ESTER-NE or ACE 2007 data;
- Topics Detection, using TDT 5 data;
- Documents Retrieval, using CLEF 2003 data;
- Objects Classification (in images) using PASCAL VOC data;
- Similar Images Retrieval, using ImageCLEF data;

3.4 The presentation framework

This last layer will ensure the standardisation of the user interfaces of the application upon the platform. This covers three different aspects: graphical aspects, integration methodology and standardization of communication between user interface fragments. As part of the integration in the global architecture the presentation framework will rely on a web interface hosted on a web portal and each pieces of interfaces will be integrated using common technologies.

Graphical aspects

The graphical aspects will enable the use of coherent graphical recommendations to identify functionalities and/or information presented to the user. Since every application can have its own graphical identify which will define the set of colours, shapes and guidelines to represent different part of the interface.

Thus, the standardisation of graphical aspects will recommend using standardized and recognized technologies in order to implement the fragment of interfaces such as the following W3C standards like HTML and CSS.

However to the inherent flexibility of these languages to structure information, extra guidelines will define best practice in the use of these languages in order to attach precise semantic to some meaningful elements of interfaces such as : content extracted from document and/or knowledge base ; generated content ; functionalities that manipulate content... The precise definition of these elements and the way their presentation will be implemented, are currently under specification, including naming conventions to identify the different parts of the interface. The use of supplementary W3C proposal, such as RDFa, will be investigated as part of this specification.

Integration methodology

The graphical user interface will be exposed to the user through a Web portal relying on the Portlet specifications (namely JSR168 and JSR286). These recommendations define the specific life-cycle of user interface as well a communication between interface elements

through the event mechanism. Originally proposed for the standardisation of interface elements in JAVA, the Portlet specifications have been extended with WSRP (Web Service for Remote Portlet) which enable the integration of non-JAVA interfaces that are remotely exposed as a specific Web Service compliant with WSRP.

Standardization of communication

The second aspects will be the use of a specific and standardised communication layer on the user interface level in order to enable exchange of information between the heterogeneous user interface fragments that comes from multiple providers.

The Portlet specification and the event mechanism already provide a technical solution for the production and consummation of messages as well as a way to define multiple types of messages. The framework will define a common typology of possible messages in order to address the multiple levels of information exchanges. The specification of these messages will follow the philosophy used in the processing framework to define generic and standardised services interfaces. Thus each interfaces elements will be organized depending on the data they can consume and/or produce as well as the functionalities they offer. Content of messages will re-use the data exchange model for the exchange of content and semantic standards (such as RDF/XML) for the exchange of high level information coming from the knowledge base.

As for the others frameworks the specifications proposed in at the presentation level will allow to ease integration and composition of multiple components for the design of application.

4 Conclusion

In this paper we described the general approach we designed to tackle the technical needs of the VIRTUOSO project.

Not only technical, the VIRTUOSO project is mainly a professional-oriented project. Over the course of the project, a community of European intelligence specialists and analysts has been gathered in a series of recurring workshops, where inputs are taken from the end-users. Every architecture detail, every innovative tool is confronted to the real-life needs of the end-users community. This community is a premiere in the field addressed by the project.

Another goal of VIRTUOSO is to go beyond existing integration platforms such as UIMA (Ferrucci and Lally, 2004), LinguaStream (Bilhaut, 2003), and Gate (Cunningham et al., 2002) (specially designed for the processing of textual documents) by (i) the use of recognized standards (RDF, XML, BPEL, etc..), (ii) processing multimedia documents, (iii) provide features for technical evaluations and (iv) provide mechanisms for reasoning on extracted information.

References

Giroux, P., Brunessaux, S., Brunessaux, Sy., Doucy, J., Dupont, G., Mombrun, Y., Saval, A. (2008). *Weblab : An integration infrastructure to ease the development of multimedia*

processing applications. In International Conference on Software and System Engineering and their Applications, ICSSEA. Paris

Ferrucci D. and A. Lally, A. UIMA: an architectural approach to unstructured information processing in the corporate research environment, 2004 Natural Language Engineering 10, No. 3-4, 327-348.

Bilhaut F., The LinguaStream Platform 2003 Proceedings of the 19th Spanish Society for Natural Language Processing Conference (SEPLN), Alcalá de Henares, Spain, 339-340.

Cunningham H., D. Maynard, K. Bontcheva, V. Tablan. GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. 2002 Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02). Philadelphia.

Acknowledgment

Funding by the Virtuoso[2] Project (FP7-SEC-GA-2009- 242352) is kindly acknowledged.

[2] http://www.virtuoso.eu/