A NOVEL HYBRID APPROACH TO SEMANTIC INTEROPERABILITY FOR LOGISTIC APPLICATIONS

Alessandro Pierpaolo Beneduce[(a)], Mosè Gallo[(a)], Vincenzo Moscato[(a)], Antonio Picariello[(a)], Lucio Sansone[(a)]

[(a)]University of Naples "Federico II"

[(a)]a.bendecuce@studenti.unina.it, [(a)]mose.gallo@unina.it, [(a)]vmoscato@unina.it, [(a)]picus@unina.it, [(a)]sansone@unina.it

ABSTRACT

As manufacturing and commerce become ever more global, companies are dependent increasingly upon the efficient and effective sharing of information with their partners, wherever they may be. In this paper, we propose an architecture that, exploiting the Semantic Web technologies, has the objective of allowing semantic interoperability among software agents for logistic applications that preserves, not only the semantic of transmitted messages, but also the subjectivity of agent's world vision in the communication. Such an architecture takes advantage by a particular communication model called Semantic Triangle in which communication agents share the referents (real world objects) and not the references (mental image or impression of a real object from the sender agents point of view) ,thus ensuring an effective semantic interoperability in the information exchange process.

Keywords: semantic web, semantic interoperability

1. INTRODUCTION

In the Internet era, large-scale computer networks and the pervasive World Wide Web infrastructure have largely solved the problem of providing ubiquitous access to any kind of information. However, while information can now be easily retrieved and accessed, the problem of processing and interpreting its meaning by automatic approaches has not yet been solved adequately and it remains an important research topic.

In such a context. as manufacturing and commerce become ever more global, companies are dependent increasingly upon the efficient and effective sharing of information with their partners, wherever they may be. Leading manufacturers perform this sharing with computers, which must therefore have the required software to encode and decode the associated electronic transmissions. Because no single company can dictate that all its partners use the same software, standards for how the information is represented become critical for error-free transmission and translation. The term *Semantic Interoperability* is frequently used to refer to this error-free transmission and translation.

We could image a *virtual supermarket* logistic scenario in which a customer (e.g. web or 3D application) asks one or more providers for a generic product with specific features. It is necessary for provider information systems (e.g. accessible by web services or REST services) understand in the correct way customer request and respond by a list of products that satisfy user needs, thus ensuring an effective semantic interoperability in the information exchange process.

Generally speaking, Semantic Interoperability is referred to the ability of two or more computer systems of exchanging information and having the meaning of sent information automatically and correctly interpreted by the receiving system. Semantic interoperability can be distinguished from other forms of interoperability (such as functional interoperability between computational units that requires the interface compatibility of connected units) because it requires that the information exchange preserves the semantic of the original message.

As a consequence, a transferred message must include, in its expressive form, all the information required by the receiving system to interpret its meaning correctly whatever the algorithms used by the receiving systems (which may be unknown to the sending system).

Presently, the important challenge of semantic interoperability among systems cannot be completely met, since current document management applications have limited capabilities for structuring and interpreting documents.

With the advent of the *Semantic Web* (Berners-Lee 2001), approaches exploiting metadata or based on semantic annotations from shared ontologies (Gruber, 1993) provide the mostly used solution to the problem of extracting some kind of knowledge from documents. The result is a number of web resources with machine interpretable mark-up that can be easily managed by software agents.

The introduction of a shared ontology solves the problem of a unique interpretation of the meaning, but it still does not keep the "subjectivity" of data sources in the communication process. Moreover, the use of shared ontologies is affected by another fundamental problem, i.e. their *maintenance*: maintaining an ontology requires trusted centralized entities that periodically up-to-date concepts from the ontology.

Such a problem has limited the use of ontologies to restricted and well-defined application domains. A further problem with software infrastructures based on shared ontologies is the scalability one that is a well-known open issue of the Semantic Web.

This paper presents a novel communication approach and a possible implementation of it that, exploiting the Semantic Web technologies, allows semantic interoperability among software agents in the Web, preserving not only the semantic but also the subjectivity of the agent's world vision in the communication. The approach is inspired to the *Semantic Triangle* communication model presented by Odgen and Richards (Odgen and Richards 1989), where communication agents share the *referents* (real world objects) and not the *references* (mental image or impression about a real object from the sender agent point of view), thus ensuring an effective semantic interoperability in the information exchange process.

For evaluating strengths and weaknesses of the proposed communication model, an instantiation of the communication process based on *semantic machines* deployed on a distributed software architecture has been developed and a case study for a logistic application has been reported

2. THE SEMANTIC TRIANGLE MODEL

Odgen and Richards proposed a model of communication between agents that characterizes each message by three distinct entities, i.e., mental thought (*reference*), sign (*symbol*), and real object (*referent*).

A reference and a symbol are linked by a symbolization relationship, reference and referent are linked by a reference relationship, while symbol and referent are not linked by any direct relationship. The relationship between symbol and referent is mediated by the subjective mind of the person who encodes or decodes the message, and thus this relationship is variable and subjective. These relationships were modeled by Odgen and Richards by the so-called *Semantic Triangle Model*.

For better understanding the introduced concepts at the base of our work, the current section will be concluded with a simple example explaining the fundamentals of communication in the semantic triangle model.

Let us suppose that Bob asks his friend John (that has a electronic devices shop) if he still has the mouse that yesterday he sold to Maria, by composing the message: "Do you have a mouse?". Because the "mouse" word (symbol) has two meanings related to two different classes of real objects (referents) - i.e. computer device or animal -, John may not understand the exact sense of the word.

Moreover, even if John understands the right sense of "mouse", he has to remember that Bob does not refer to a generic mouse, but to the particular one that yesterday he sold to Maria. Thus, the correct communication outcome depends on the image that the "mouse" word creates in John's mind (reference). In order to preserve the semantic, Bob has to attach useful information (reference) to the message - e.g. by composing the message "Do you have the mouse that yesterday you sold to Maria?" - that permits John to understand what is the real object (referent) that Bob asked him, although John has a different personal concept of "mouse". In fact, the reference "that yesterday you sold to Maria" can be seen as a sort of pointer to a local but sharable concept of Bob's memory.

3. APPROACHES FOR MANAGING THE SEMANTIC INTEROPERABILITY

Selvage et al. outline in (Selvage 2006) different architectural patterns for achieving semantic interoperability in distributed environments and place at the opposite extremes of their list the *point-to-point semantic integration pattern* and the *Semantic Web* pattern.

In the point-to-point semantic interoperability pattern, the communication is based on messages which directly embed a complete description of their semantic. This aspect implies that the message minimum semantic unit includes both symbols and descriptions of referents, which instances could be maintained in sender local knowledge sources (e.g. ontologies).

This approach represents a *subjective* communication approach where the subjective vision of the sender knowledge is preserved, but information redundancy and incoherency problems may arise.

Vice-versa in the Semantic Web, based on the *Semantic Web Layer Cake* (Berners-Lee 2001)model, communication relies on the use of ontologies that are conceptualizations of specific domains in order to formalize semantic of data. The Semantic Web links and relates elements of a message to a common ontology, using the *Resource Description Framework* and the *Web Ontology Language* that allow data to be shared and reused on the Web. A symbol will be directly linked to its referent by means of an ontology which provides the correct semantic of real world objects, thus solving the problem of message ambiguity.

This kind of communication approach can be classified as an *objectivistic* one, where the knowledge (that is formalized by the ontology) is independent on the agents involved in the communication. This model obviously simplifies the communication problem and the implementation of systems based on such an approach, but some new problems: *ontology acceptance*, *ontology building* and *maintenance* and *ontology expressiveness* (Devedzic 2001, Harris 2004, Gangemi 2005).

In order to cope with weaknesses of both communication approaches, it is possible to propose a sort of *hybrid model* that exploits the interoperability mechanisms of both the subjective and objective model. In particular, in the hybrid model each sender agent has its own subjective knowledge that may be either mapped into shared objective knowledge sources (such as an ontology), or directly included in a coded message in order to preserve its personal interpretation of transmitted concept, coherently with Odgen and Richards model.

4. AN IMPLEMENTATION OF THE HYBRID MODEL

Internet and the Semantic Web offer the necessary infrastructure for implementing the hybrid communication model presented so far, and obtaining a semantic interoperability among software agents in the Web (Hendler 2001). In this section, the software requirements of a possible implementation of the communication model will be presented. In particular, the characteristics of transmitted messages and the communication process involving them in the hybrid communication model will be presented.

4.1. The Message Conceptual Model

Messages exchanged in the hybrid communication model are considered as the aggregation (*information objects*) of *digital assets* (e.g., images, textual documents, audio, etc...) containing a set of *information concepts* that constitute the message semantic content that an agent is interested to transmit; these concepts refer to the agent knowledge that can be mapped either into a local ontology, or a universal ontology that is a-priori accepted by all the agents involved in the communications.

The information concepts can be classified in two distinct types: *entities* and *facts*. An entity is a noun, verb or other part of a speech that can be retrieved on any language dictionary. A fact is an expression corresponding to peoples, places, events or any other thing that could not be retrieved in a language dictionary. Facts can further be classified into *Encyclopedic* and *Non-Encyclopedic* facts.

Encyclopedic facts have a general relevance such that they could be contained in a general encyclopedia or in a domain encyclopedia. Referents to encyclopedic facts can be called *Universal Fact Referents*, since they have to be universally shared and accepted by any communication agent (or at least, they have to be shared by communication agents belonging to a specific domain).

Non-Encyclopedic facts are relevant in the internal world of the agent but could be not universally relevant or there could not be a universally accepted semantic. Referents for non-encyclopedic facts are named *Local Fact Referents* and are maintained in the Semantic Machine of the agent.

To make explicit the binding of information concepts with real word objects referents, we propose to use *semantic tags* for providing a symbolic representation of information concepts, and references for binding together semantic tags and related referents

4.2. The Communication Process

The proposed communication process is supposed to be decomposed into four sequential activities:

1. *Information encoding* - a sender agent composes a message in a particular format that is understandable by other semantic machines (in this step the binding between information concepts and referents is performed using semantic tags and the reference mechanism;
2. *Information transmission* -sender agent publishes the message (e.g. by web pages) on the web, or transmits the message to a receiver agent (e.g. by e-mail);
3. *Information acquisition* - a receiver agent performs the message acquisition (download);
4. *Information decoding* - receiver agent binds the received information with the related referent by means of sender references.

With respect to the example discussed in section 2, the communication could be outlined as follows:

1. Bob uses an e-mail client (semantic machine) to compose the message “Do you have a mouse?” that is represented in a particular format (e.g. XML) and in which the personal concept of a “mouse” is bound to the related referent by the tag “mouse” and a reference (e.g. an hypertextual link pointing to a web page deployed in a web server containing the description of the mouse);
2. the e-mail is sent to the pop server of receiver;
3. the email is downloaded by John;
4. the semantic machine of John decodes the message and binds the concept of “mouse’ to the corresponding referent using Bob’s reference; after the communication, John may decide to add it to his local ontology for ordering a new mouse of the same kind to his provider.

The Message Sending and Receiving activities are specified by the UML Activity Diagrams reported in Figures 1 and 2. In particular, Sending a message includes the *Tag Extraction* and *Reference Binding* sequential activities (implementing the message encoding), besides the *Encoded Message Transmission* one.

The *Tag Extraction* activity performs the retrieval of information concepts from a message to be transmitted and their symbolic representation by semantic tags. The set of these semantic tags can be indicated directly by the message author or it can be retrieved automatically by tag/information extraction algorithms.

For each tag, the *Reference Binding* activity is performed in order to retrieve from the local ontology a reference to the information concept represented by the semantic tag. More precisely, for each tag the *Local Ontology Mapping* activity is executed, in order to search in the local ontology for possible references associated with the tag.

If more than one possible reference is found, then a *Local Disambiguation* activity is carried out, where a single reference correctly representing the tag has to be selected. After the eventual disambiguation, two cases may happen:

1. the reference correctly representing the tag has been found in the local ontology (*Mapping Hit*) and it is possible to bind the message tag with this reference (*Bind Tag with Reference* activity);
2. the tag has not a corresponding reference in the local ontology (*Mapping Miss*), thus, a *Referent Search* activity is entered, where a referent reporting the semantic of the information concept must be found in an available Referent Source, and therefore a corresponding reference (*Create Reference*) has to be built.

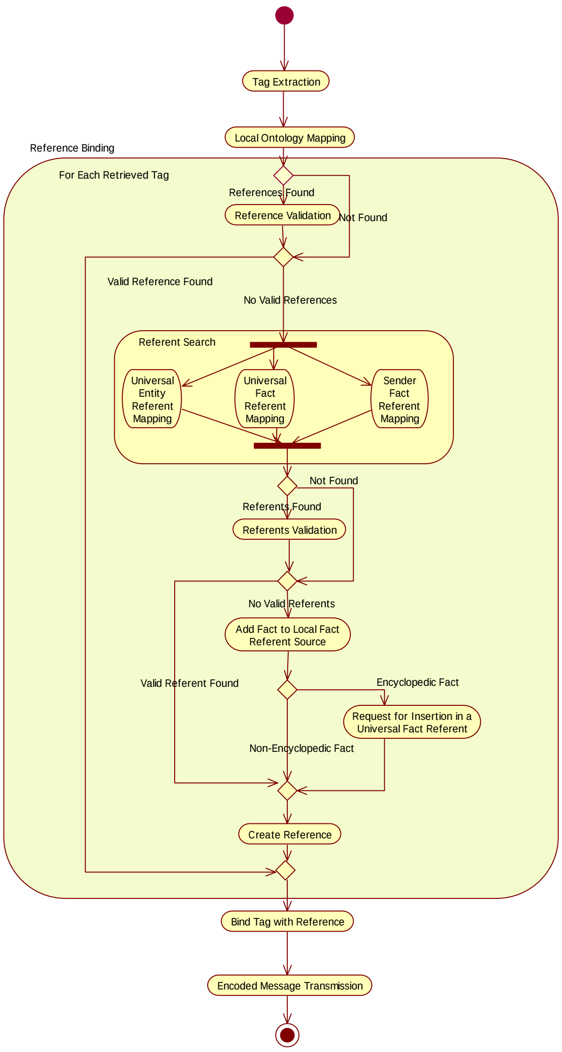

Figure 1: The Message Sending activity diagram

In the Referent Search activity, different sources can be queried depending on the type of information concept to be represented and on the available Referent Sources.

As an example, Entity Referents can be retrieved in Universal Entity Referent Sources, such as *Wordnet* or other on-line dictionaries. Encyclopedic relevant facts can be retrieved in Universal Fact Referent Sources, such as *Wikipedia*. Encyclopedic relevant facts related to a specific domain can be retrieved in sources containing domain ontologies. Non-Encyclopedic relevant facts, i.e. facts that are relevant only in the sender context, can be retrieved just in a Fact Referent Source published by the sender itself.

Facts for which the sender declares a semantic that is different, or more specific, from the one proposed by a universal fact source, can also be retrieved in the Local Fact Referent source of the sender.

In our architecture, each universal referent is a directly addressable resource. It can be accessed either by a HTTP GET service request, or by a web service that performs the wrapping of the related referent source [6]. For accessing local referents (also coded in XML/RDF) a possible solution is offered by the REST technology. In this case, a referent can be directly stored into the Local Fact Referent Source of the semantic machine via a HTTP PUT or POST request, and directly accessed via a HTTP GET request.

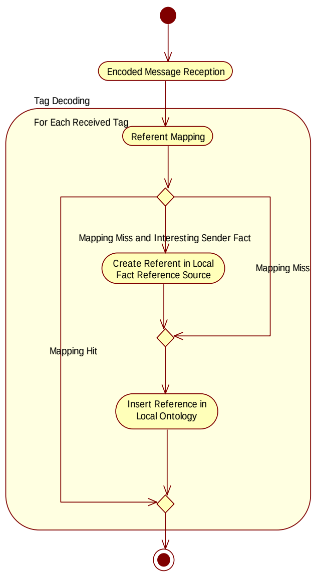

Figure 2: The Message Receiving activity diagram

At the end of the Referent Search activity, it is possible that more than one possible referent has been found. In this case, a *Validation* activity is needed, in order to state if any of the referents provides a satisfying semantic for the information concept, or to select the more suitable one, if more than one referent was found. If a satisfying referent has been found, then the *Create Reference* and *Bind Tag* activities are performed, and a new reference to the chosen referent is added to the local ontology and will be bound with the tag.

Elsewhere, a new referent has to be inserted in the Local Fact Referent source. Eventually, a *Request for*

Insertion activity for adding the new created referent to the Universal Fact Referent Source can be performed. This possibility is granted by some sources, such as Wikipedia, but the referent will be inserted only if it will be accepted by the referent source managers, and only in an asynchronous way.

At the end of the Reference Binding activity, the message minimum semantic unit will be composed by a set of couples <*tag, reference*>.

In our architecture it is considered as a web resource characterized by an URI with naming and addressing functionalities, and will be encoded by XML/RDF languages. Thus, using its URI, each information object can be accessed by HTTP protocol via a GET request, and the result of such a request will be an XML/RDF representation that separates the physical view from the resource view of contents, thus reflecting the semantic web layer cake schematization, and providing interoperability with applications of the Semantic Web. As to the receiving activity, its purpose is to map each received couple <*tag, reference*> into any concept from the receiver agent local ontology, in order to reconstruct the correct semantic (Referent Mapping Activity). If a reference is found in the local ontology (Mapping Hit), the information concept is known to the receiver and it must not be decoded. Otherwise, if no reference is found, then a new reference has to be created and inserted in the local ontology. Moreover, if a received concept points out from the sender fact referent source, then the receiver agent can choose to accept the referent and to add it to its Local Referent Fact source.

4.3. The Semantic Machine

The described communication process can be implemented in distributed environment including several *Semantic Agents*, each one equipped with a *Semantic Machine* that executes message sending and receiving activities. The realization of the Semantic Machine will require a number of design decisions, and will depend on several technological choices. A reference software architecture of the Semantic Machine is outlined by the UML Deployment Diagram reported in Figure 3, and a description of its five logical components is provided in the following.

Encoder is the Semantic Machine component responsible for implementing the information encoding and transmission activities. It asks the *Ontology Manager* component for retrieving/inserting references in the local ontology and asks the *Referent Manager* component for referents in the available Entity or Fact sources, and for inserting referents into the Local Fact source.

Decoder is the component responsible for the information acquisition and decoding activities. It queries the Ontology Manager to retrieve/insert references in the local ontology and asks the Referent Manager for inserting new referents into the local Fact source.

Tag Extractor is the component responsible for the automatic or human-assisted extraction of tags representing the relevant semantic concepts of the information object.. The extracted tag list is returned to the Encoder component for the elaboration.

Ontology Manager is the component responsible for managing the local ontology, offering reference searching and reference inserting services. The implementation of the Ontology Manager will depend on the technologies and languages adopted for the realization of the Local Ontology. Possible languages can be RDF or OWL , while apposite API provided by open-source tools (e.g. JENA) can be exploited to create, manage and query (via SPARQL) the set of references.

Referent Manager is the component responsible for managing the referent sources, offering referent searching and referent inserting services. Of course, the implementation of such a Referent Manager will depend on the technologies used to encode the referent sources.

Using the Internet infrastructure, the ***Universal Referent Sources*** can be provided by web sites or any other data source exposed on the Web. Some sources (e.g. *eBay* or *Wordnet*) offer APIs for providing direct access to the referents, while other sources do not provide them, and thus wrapping techniques (Canfora 20008) are needed to access referents. The ***Local Fact Referent source*** is deployed in the semantic agent and it can be developed by using local database or XML documents.

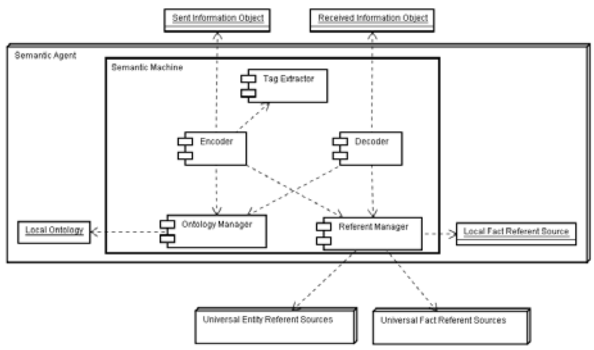

Figure 3: UML Deployment Diagram of the Semantic Machine

5. A CASE STUDY

In this section, the implementation of a semantic machine that supports the execution of the hybrid communication process in a specific context will be presented. We use as case study of the proposed semantic interoperability model a virtual supermarket scenario, where costumers (sending agents) during a purchase of a movie DVD can transmit on the web a message related to the desired product, while providers (receiver agents) give an interpretation to messages and respond (sender agents) with the description of a product that try to match user needs. Costumers (receiver agents), eventually, can decide to buy a given product.

For concepts related to the movie domain, it is a commonly accepted opinion that the *imdb.com* database (accessible via the *www.imdb.com* website) is the larger and reliable source of information available on the Web. Thus, in the case study imdb.com has been selected as a referent source for facts such as movies, directors, actors, cinema events and so on. For all the other facts, i.e. facts that are encyclopedically relevant but not directly related to the movie domain, the English version of the *Wikipedia* web-site has been selected as referent source. Since most of the musical concepts are also retrievable on Wikipedia, a disambiguation activity was needed in some cases.

As to the entities, the *Wordnet* repository has been considered as the unique Referent Source for entities in English language. Finally, for all the facts that are not encyclopedically relevant (i.e. for which no valid Referents can be retrieved in the considered Referent Sources) or for which the referent source is considered not reliable or not detailed enough, a Local Fact Referent Source deployed in the semantic agent was considered.

5.1. Working Example Execution

We assume that a costumer semantic agent wants to send to vendor semantic agents the digital asset consisting of the following text fragment:

"*I would like to see an action/adventure movie. I like gothic atmosphere with vampire and werewolves. I love the Dracula novel and I like actors as Keanu Reeves, Monica Bellucci and Winona Ryder and Anthony Hopkins and directors as Francis Ford Coppola* ".

The editor provides the following set of tags related to the digital asset: 'action movie' - 'adventure movie' - 'vampire' - 'werewolf' - 'gothic' - 'Dracula' - 'novel' - 'Keanu Reeves' - 'Monica Bellucci' - 'Winona Ryder' - 'Anthony Hopkins' – 'Francis Ford Coppola'.

During the encoding process, corresponding referents for each tag have been searched in the known referent sources, obtaining the following results:

1. added to the local fact source: nothing ;
2. retrieved on wordnet and wikipedia: 'novel', 'Dracula', 'vampire', 'werewolf';
3. retrieved on wikipedia: 'action movie', 'adventure movie';
4. retrieved on wikipedia and imdb.com: 'Keanu Reeves', 'Monica Bellucci', 'Winona Ryder', 'Anthony Hopkins' , 'Francis Ford Coppola'.

The result of the encoding process was a XML/RDF document reporting the list of tags with the corresponding URIs of chosen referents that vendors have to decode to respond with their products.

6. CONCLUSIONS

In this paper, we proposed an architecture that allows semantic interoperability among software agents for logistic applications. Future works will be devoted to implement wrapping modules for other referents sources for and realize a complete experimentation in order to obtain significant results

REFERENCES

Berners-Lee, T., et al., 2001. The Semantic Web, *Scientific American*, 284:34-43.

Canfora, G., Fasolino, A.R., Frattolillo, G., Tramontana, P., 2008. A wrapping approach for migrating legacy system interactive functionalities to service oriented architectures, *Journal of Systems and Software*, 81(4):463-480.

Devedzic, V., 2002. Understanding ontological development, *Communications of the ACM*, 45:136-144.

Gangemi, A., 2005. Ontology design patterns for semantic web content, *In ISWC 2005 (LNCS 3729)*, pag. 262-276.

Gruber, T., 1993. A translation approach to portable ontology specifications, *Knowledge Acquisition*, 5(2):199-220.

Harris, J., 2004. Judging the likely success of an ontology, *Technical report*

Hendler, J., 2001. Agents and the semantic web, *IEEE Intelligent Systems*, pages 30-37.

Ogden C.K., Richards, I.A., 1989. The meaning of meaning, *Ed. Harcourt Brace Jovanovich*.

Selvage., M., et al., 2006. Achieve semantic interoperability in a SOA - patterns and best practices, *Technical IBM report*.

AUTHORS BIOGRAPHY

Alessandro Pierpaolo Beneduce received his Bachelor degree in Logistic Management Engineering from University of Naples "Federico II". Actually, he is collaborating with the Computer Science Department for the development of 3D logistic and e-commerce applications.

Mosè Gallo received his Master Degree in Mechanics Engineering and the Ph.D. degree in Logistics from University of Naples "Federico II". His research activity concerns logistic modeling and simulation.

Vincenzo Moscato is san assistant professor of Database at University of Naples "Federico II". Actually, his main research interests are Multimedia Database and Semantic Interoperability.

Antonio Picariello is san associate professor of Database at University of Naples "Federico II". His research interest lie in the fields of Multimedia Database and Knowledge Management.

Lucio Sansone is a full professor of Information Systems at University of Naples "Federico II". His research interest lie in the fields of Multimedia Information Systems and Business Process Reengineering.