# Towards a Semantic Contact Management

Irene Celino, Francesco Corcoglioniti, and Emanuele Della Valle

CEFRIEL – Politecnico of Milano, Via Fucini 2, 20133 Milano, Italy
{irene.celino,francesco.corcoglioniti,emanuele.dellavalle}@cefriel.it

**Abstract.** Many organizations face every day the problem of effectively managing their contacts (customers, suppliers, partners, etc.), in terms of communication, clustering, networking, analysis, and so on.
Our company decided to cope with this issue by gathering the requirements for Contact Management and by designing and developing a prototype, called GeCo, to fit Cefriel needs.
During the development of this application, which ran in parallel with some research projects dealing with Semantic Web technologies, we recognized that the addition of some "semantics", both in the data modeling and in the tool design, would help a lot in solving the open issues for the general problem of Contact Management.
In this paper we summarize the main criticalities in managing contacts and we suggest how Semantic Web technologies can contribute to their successful solution.

## 1 Introduction

Today, the success of an organization considerably depends on its ability in managing and cultivating its network of contacts, namely customers, suppliers, partners.

The more the organization grows, the bigger this network becomes; moreover, it becomes difficult to have a unitary view on that network, because the different organizational units have different demands which lead to the adoption of different solutions of Contact Management. Information about contacts, as a result, is fragmented within the organization or duplicated between organizational units, each one with its different view. This situation can cause inefficiencies (e.g., the problem of "who knows who" arises) and loss of opportunities (e.g., a unit cannot exploit acquaintances of other units of the same organization, because their contacts are inaccessible).

A solution can come by managing contacts in a (logically) centralized way, by a collaborative approach in which each member of the organization can edit and access all the contacts. This way, each contact is associated to a rich profile, obtained from the aggregation of information coming from several sources, primarily from users but also from legacy systems and other, pre-existing sources. This rich knowledge base, no more fragmented, provides a unitary and consistent view of the contact network. There are a lot of advantages: improved efficiency and better data quality (because of the shared data entry and management), improved search of contacts (e.g. by expertise area), no opportunity loss (since all data is now easily accessible), enabling advanced operations otherwise difficult to achieve, such as analysis to support strategic decision and planning (e.g., "what existing contact could be interested in the product X?").

At Cefriel (our company), we designed and developed a prototype of Contact Management application following this centralized and collaborative approach. In the beginning we chose "traditional" technologies, but several difficulties came out and led us to speculate on a Semantic Web solution. On the basis of the preliminary results, in this paper we present the challenges and problems that arose in developing such a Contact Management application and we investigate the applicability of approaches and technologies from the Semantic Web world to solve the open issues.

The remainder of the paper is structure as follows: section 2 outlines what Contact Management is and how it is declined in our company; section 3 analyzes the problems and issues that arise when realizing an application to deal with Contact Management, independently of the technologies chosen; in section 4, we envision how the employment of Semantic Web technologies can help in solving the issues presented in the previous section; finally, we draw some conclusions and illustrate the following steps in section 5.

## 2 Contact Management

Contact Management deals with the acquisition and maintenance of knowledge about a user's contacts, where the term *contact* could be given the broad meaning of *someone who the user has been exposed to during communication* (see also [1]). Contact Management is a composite discipline which deals with different other approaches and demands, like:

- *Personal Information Management* or PIM, which is about keeping trace of personal contacts, together with details about when and where the acquaintance was born and the events and projects in which those contacts were cultivated; PIM systems are often the first step towards the sharing of those information within specific groups (with an opportune access control);
- (Analytical) *Customer Relationship Management* or CRM, which is about the management of customers/partners, their interests, their past purchases, closed/activated/proposed projects with them and the analysis of the contacts to support the business strategy of the organization;
- *Public Relations* or PR, which are about the timely and customized communication with the stakeholders outside the organization.

Cefriel[1] is a private not-for-profit ICT company with about 130 employees with different specializations and expertises, which deals every day with a number of different interactions with various stakeholders. Cefriel's mission is realized by three kinds of projects:

[1] http://www.cefriel.it

1. Research Projects, at national and European level, which require the interaction with partners and evaluators (e.g., the European Commission delegates);
2. Innovation Projects, with industry as well as public administrations, aimed at innovative solutions and technology transfer, by identification, analysis, evaluation, integration and creative use of off-the-shelf and new technologies;
3. Educational Projects for post-graduates students as well as training courses for ICT professionals and executives.

Within those projects, the main needs for Contact Management comprise the maintenance of relationships with stakeholders and the search of external contacts or internal staff, primarily by expertise area, in order to launch a new activity or to manage a current one.

In order to deal with all those heterogeneous contacts, we decided to gather all the different requirements for their management and to design and develop a prototype of Contact Management application. We named this tool GeCo (from the Italian "*Ge*stione *Co*ntatti", i.e., Contact Management) and we employed traditional technologies to build a web application which allows the collaborative editing, update and retrieval of the contacts, as well as the creation and management of groups and distribution lists.

## 3 Issues about the implementation of a Contact Management application

While designing and developing GeCo, we tried to realize a logically-centralized and collaborative solution; by doing this, we met several issues that we can schematize under five macro-categories: *data acquisition*, *data relevance and recency*, *extensibility* (of both the model and the tool), *data fruition*, *privacy and security*. In the following, we analyze each one of those categories, highlighting the main difficulties we met in solving the problem of Contact Management from a (traditional) technological point of view.

### 3.1 Data Acquisition

The acquisition and the preservation of contact information are expensive and delicate activities, since contact data must be often inserted manually and the correctness of data is crucial to assure the usefulness of the Contact Management application (e.g., there must be no mistake in emails or telephone numbers).

We can identify two principal ways for acquiring contact information, namely manual editing and reuse of pre-existing data sources (internal or external). None of these is sufficient on its own and both the approaches are required for Contact Management. For example, there are situations where manual editing is inappropriate or disallowed, since contact data could be already available (e.g. in the company LDAP directory) or it can also happen that organizational reasons prevent some information (e.g. employees' data) from being edited in a collaborative way. Vice versa, it is frequent that required information must be explicitly provided by users, due to the absence of existing data sources.

In GeCo – the tool we developed – manual editing is designed to be a collaborative activity, since each user can insert and modify contact data. This approach makes the solution stronger, enabling the optimization and a better distribution of the contact management effort, that is delegated to the contributing users.

The collaborative approach is undoubtedly one of the greatest strength of the GeCo solution, since a contact inserted by one user can be re-used by all other users of the system; nonetheless, this "teamwork" editing makes hard to assure consistency and quality of the inserted data. For example, we can meet the following problems:

– *Contact Identification and Uniqueness*: the same contact can be inserted multiple times by different users which don't realize that the specific information is already present and stored in the system;
– *Concurrent Updates*: different users editing the same contact can disagree on the shared data (e.g., the value of one or more contact properties); there is also a strong need of versioning and configuration management, because a user can inadvertently and wrongly overwrite previously correct data, causing the need of reverting the data to a previous version.

Regarding the reuse of existing data sources, we have to face the typical problems of data integration, i.e. the heterogeneity of data and schemata and the presence of duplicated, inconsistent or incomplete data. Although marginally, these problems arose also during the implementation of the GeCo prototype, when we had to integrate existing employees' data coming from several sources.

### 3.2 Data Relevance and Recency

In Contact Management systems, a perceived problem is assuring the relevance of the inserted contacts and the recency of related data.

Contact Management is an expensive task: the greater the number of managed contacts, the more expensive their acquisition and preservation, and the more difficult the functionalities of search and fruition over them. Therefore, given that different interlocutors of a person or different partners of an organization are not equally important, it is a good practice to keep track only of the relevant contacts in order to reduce the maintenance effort.

Nonetheless, identifying the relevant contacts is not a simple task, because they can change over time: an interlocutor or partner, which is important today for a running activity, can become less interesting for the company when that activity ends. As a result, whenever facing the decision of evaluating the relevance of a contact, users are inclined to go for the most conservative choice, by keeping memory of irrelevant contacts that will never be deleted. As an example, a test conducted between users with different functions and various communication demands highlighted that only 19% of stored contacts was actually relevant [1]. This fact was noted also during the deploy of GeCo and we took advantage of the initial loading for "cleaning" the data from irrelevant contacts.

Another relevant aspect in managing contacts is data recency. It must be noted that a significant part of contact information has a dynamic nature (e.g., a person can change his telephone number, move to a different address, leave his occupation for a new job, etc.). Therefore, it is crucial to have up-to-date contact information.

This requirement has a strong impact over the management of contacts, especially when data is inserted manually by users. How can the system keep the information always up-to-date? How can a user, when accessing some contact data, verify whether those data are still valid? This problem arises also when a single person is responsible for the contact management, but it is more strongly perceived when the approach to the data editing and update is collaborative, because data could be inserted by a different - and perhaps unknown - user (e.g., who inserted this piece of information? can I trust that editor?).

### 3.3 Tool and Model Extensibility

Even in a middle-sized company, we can identify different needs in the Contact Management, coming from different classes of users. Directory Board, Communication, Marketing, Technical Support: those units have different requirements and are interested in different kinds of information when dealing with contacts.

A widespread solution consists in the use of different, independent tools that respond to specific demands. This, however, leads to data fragmentation which, in turn, hampers the development of new opportunities and the efficiency deriving from a unitary and overall vision.

On the other hand, by adopting a logically centralized solution, the different requisites must be supported by a single tool, enabling each class of users to store and manage the contact information of its interest. This means that the tool, for each contact, will manage:

1. a *baseline set of properties*, which are in common for the various users (e.g., name, affiliation, telephone, email, etc.);
2. various sets of *additional information*, to support specific demands, to be handled by different applications or to be used by different classes of users (e.g., participation to projects or training courses, interest in specific products or services offered by the organization, etc.)[2].

Many requisites are typically not apparent or clear during the design phase and often originates only after the adoption of the tool. As a consequence, the tool and the underlying data model have to be extended in order to meet the new requirements, with the risk of complex adaptations or refactorings, perhaps justified by the advantages of an integrated contact management approach. For those reasons, the Contact Management tool is required to adopt an easily extensible architectural solution, in terms of data schema and tool functionalities, so that acquisition and access to additional data can be provided at "run-time", when the application is already in the production environment.

[2] Particular care must be taken, however, when adding functionalities to a Contact Management tool, in order to prevent the "overloading" of the tool with the business logic specific of conceptually different tools.

To complicate things further, to extend the model or the tool it is often necessary to integrate or link different information sources or other internal applications (e.g., a product DB), which often are pre-existing and cannot be modified (e.g., when they are strictly connected to the core business of the company). Nonetheless, in order to fulfil the necessary functions, the tool must be able to integrate data from those sources, or to put the contacts in connection to information stored in the sources.

### 3.4 Data Fruition

Once we gathered a large amount of complex data which respond to the different requirements, and once we found a solution to manage their consistency and update, we still have to enable an effective and efficient fruition by the final users of the Contact Management system.

In this regard, we can identify three main requisites:

1. *Contact Organization*: the different users have specific demands in organizing their contacts, for example they wish to classify or to group them in categories, or they need to define lists of addresses for specific objectives, like distribution lists for advertising. In the most common Contact Management tools, there is a little support for contact organization, because the users operate in different "fluid" contexts, in which teams, activities, projects and associations change over time.
2. *Contact Search*: when the number of contacts is very high, it can be difficult to perform effective searches over them. We can further distinguish between two kinds of search:
   – *Formal or Directed Search* (cf. also [2]), when the user is looking for a specific contact of which he knows, for example, name and surname; the input information can be incomplete or lead to multiple contacts selection (e.g., a lookup for "John Smith" which results in a hundred matches): the tool must ease the individuation of the "right" result (e.g., by enabling a search refinement);
   – *Free or Indirected search*, when the user is looking for contacts with specific characteristics (e.g., all the employees of company X, all the contacts whose birthday is tomorrow, etc.); a frequent and crucial need is the search of contacts by expertise area (see also [3]).
3. *Access or Reuse of Contact Information*: especially in the everyday use of PIM tools, it frequently happens that the user wishes to reuse a selected contact, for example by importing it in the address book of his mail client or by using directly its email address to send a message. The integration between the Contact Management system and other PIM tools (like the mail client in the previous example) must be two-way, i.e. users should be also allowed to import contact information into the Contact Management tool and integrate it with the possible pre-existing data, in order to avoid replication or fragmentation of data over multiple tools.

### 3.5 Privacy and Security

Contacts are often a valuable asset of an organization and as such they should be properly protected. Typically, contacts' data may comprise confidential information and its analysis could reveal important details about the organization, its relationships and strategy.

As a consequence, access to (part of the) contacts or related information is often restricted by security policies. For example, confidential telephone numbers and personal information can be accessible only to selected users, or some information, though public and not confidential, is editable only by a subset of users, since it is strictly bound to specific demands. Moreover, it can be useful to let users associate notes to a contact (e.g., memoranda, remarks about task progress, comments and personal opinions on the person, etc.); in that case, those notes must be considered private and be accessible only by their respective author.

Security constraints are in contrast with a purely collaborative approach, nonetheless their satisfaction is essential to make organizations adopt the tool. Therefore, a Contact Management tool must define and implement an appropriate model for access control and privacy preservation and must support users in defining access policies and problem-solving guidelines (e.g., whenever a contact is duplicated by a user that had no access to the pre-existing information).

## 4 Adding "semantics" to a Contact Management application

All the issues presented in the previous section arose when we designed and developed the GeCo prototype. This experience taught us the problems, difficulties and limits, arising in realizing a Contact Management application, which, addressed with "traditional" approaches and technologies (as we initially did), prevented us from getting a brilliant and comprehensive result in the end.

However, the lesson learned from this experiment, together with our knowledge and experience about Semantic Web technologies, led us to several considerations about what would represent the best possible solution for Contact Management. We strongly believe that Contact Management could heavily benefit from the adoption of a "semantic" approach and this is the main motivation for writing this paper. Both current results from the Semantic Web field and further development of standards, methods and techniques which are envisioned in the research community can greatly improve Contact Management applications and help to achieve better results and provide new solutions to the different challenges we introduced in section 3.

In the following, we follow this direction by investigating, for each issue introduced in the previous section, the "semantic" solutions we foresee and the advantages they promise with regards to more traditional approaches. We present the effects we can get with current Semantic Web technologies and solutions (e.g. RDF, ontological models, the FOAF vocabulary), as well as the outcomes we expect from the ongoing standardization efforts in the field (e.g. SPARQL, RDFa, GRDDL, RIF); in the latter case, we try to outline our expectations and demands.

### 4.1 Semantics in Data Acquisition

A possible way to handle the consistency problem arising from manual editing (duplicated data and concurrent updates, as outlined in 3.1) is to track all the modifications occurring to contact data. More specifically, we designed the GeCo system in order to keep trace of:

- *Data Provenance*, i.e. what data were inserted by which user; this enables a user to ask directly to the author about the reliability and trustworthiness of each piece of stored information;
- *Update History*, i.e. the "log" of updates (e.g., inserted values, date, editor); this enables a correct versioning of data, with the consequent possibility of roll-back in case of erroneous data update.

Representing and managing all the data described above revealed to be quite troublesome in the architecture we chose, based on the use of relational DBs. Therefore, in order to limit the complexity, we opted for a partial solution, by keeping trace only of the last update editor and by storing within the history only the complete contact information rather that the single modified values.

However, a full solution to those problems would have been achieved quite easily by representing information by means of RDF triples [4] described by an ontology, because:

1. Data provenance and update history can be compared and re-conducted to metadata associated to triples, using Named Graphs [5] or, with proper care, reification [6]. In this way, we exploit RDF capabilities in (1) treating and representing complex information in a homogeneous way with graphs of triples and (2) working on data with a fine-grained granularity, by associating metadata to triples or groups of triples (the same technique is quite troublesome, for example, in relational databases). With regards to data provenance, we strongly support a wider adoption of provenance tracking techniques and we look forward to seeing a progress in the standardization of solutions like Named Graphs.
2. By using ontologies, we can formally describe integrity constraints and inference rules, in order to assure the coherence of the knowledge base (e.g., we can exploit rules, based on properties values, to infer the equivalence of duplicated contacts). In that respect, we foresee great improvements towards a comprehensive solution through the standardization activities of RIF Core and its dialects [7].

Regarding the reuse of existing data sources, RDF is well suited for data integration, which could be achieved through (1) wrapping external data sources by means of mappers that expose a SPARQL [8] end-point, (2) integrating the different data simply by merging RDF graphs and, in case, (3) exploiting inference rules to identify relations between different data or to state the equivalence of data coming from different sources and describing different aspects of the same resource.

Apart from the databases and legacy sources typically present in an organization, an interesting kind of source is represented by the profiles and electronic business cards published by users on the Web (e.g. in their home page). The importance of those sources is twofold: on the one hand, their use helps in reducing the effort required for contact editing and maintenance; on the other hand, they represent authoritative sources, since data is maintained directly by their respective owners and we can usually rely on its correctness and recency (as discussed thoroughly in section 4.2).

To enable the system to deal with those external sources, the information must be expressed in a standard format. In this regard, there are some proposals for standard like vCard [9], but we believe that better results can be obtained with Semantic Web approaches which allow for making data semantics explicit and therefore machine processable. Pragmatically, contact information can be encoded directly in RDF by some reference vocabulary or ontology (e.g., FOAF [10]). This information could then be published on-line as Linked Data [11] or even incorporated in the (X)HTML pages (e.g., via RDFa [12] or Microformats[3]) and extracted thereof in an automatic way (e.g., via GRDDL [13]). With this solution, each contact can be identified and referred to by a dereferenceable URI, so that the Contact Management system could (semi-)automatically access the most up-to-date information whenever needed.

### 4.2 Semantics in Data Relevance and Recency

In Contact Management, the problems of data relevance and recency are crucial. In the GeCo prototype system, we identified two possible families of solutions to manage the selection and update of relevant contacts:

1. A *technical* solution (based on the provenance data and update history maintained by the system, see 4.1) tracks the last user and the last modification date of a contact, in order to give a rough hint about data recency. Actually, this is not a comprehensive solution, since it helps only in identifying the user to ask to about the "freshness" of data.
2. An *organizational* solution can be achieved by defining proper guidelines to associate each contact to an internal responsible editor (e.g. the person who initially inserted it) to whom the update of contact information is delegated. This solution is partial too, since it relies upon users' good will to follow the guidelines; of course it would be possible to code the guideline enforcement within the system, but this would make the solution too stiff.

None of the previous solutions proved to be completely effective. A further step toward a more comprehensive solution, proposed in [1], consists in automatically selecting the relevant contacts, in order to (visually) filter out the other irrelevant ones. This selection prevents the "waste" of time and effort related to their maintenance. In the cited work, the relevance of a contact is related to different factors belonging to two categories:

– *Communication History*, which includes parameters like frequency, recency, longevity, presence of long-term interactions, reciprocity;

[3] `http://microformats.org`

– *Communication Style*, which is about the attitude of a user towards communication and contact management, deriving from his interaction demands.

The automatic selection of relevant contacts is a feature that we cannot disregard and that can be facilitated by the use of Semantic Web technologies. First of all, it enables the visual interface to hide the irrelevant contacts, in order to reduce the informational overload for the users. This result can be achieved by selecting the RDF nodes to be visualized or by querying the contacts' RDF information to extract only the desired metadata. In the second place, this feature can be the basis for advanced functionalities like alerting, filtering and prioritization for inward communication, or reminding to cultivate relationships (e.g., alerting a user to contact a person that he did not hear from recently). We can get this result by defining opportune inference rules to determine the information to be notified or suggested to the user. A related work is the Semantic Email Addressing proposed in [14], which suggest a way to address email to a targeted audience based on the "semantics" of the group of addressees.

Finally, another way to select the relevant information and to keep it up-to-date, as well as to simplify contact update and editing tasks, consists in exploiting and integrating external information sources (as introduced in 4.1), which can be either more up-to-date or authoritative, when they represent the "official" source for data about a contact (e.g., personal home pages which publishes personal data, telephone numbers, emails, etc.). This approach requires a support from outside the organization, but can achieve better results or, at least, more reliable data.

### 4.3 Semantics in Tool and Model Extensibility

The requisites we drew in section 3.3 imply the need for extending the information managed by the Contact Management tool, after its development or deployment, and in case directly at run-time.

The information extension can be applied at different levels: on the data layer, on the application layer and on the user interface level.

At *data level*, extending the information means enlarging the model, by adding new types of information and by integrating or linking to external data sources. To this end, a Semantic Web approach appears to be particularly suitable and useful: enlarging the model becomes concrete by extending the ontology (adding new concepts and properties or importing other ontologies). Moreover, integrating external data sources could be easily achieved by using RDF (see 4.1). Data integration could be interesting also in the opposite direction, by letting other systems to access information managed by the Contact Management tool; the enforceability of this approach, however, heavily depends on the characteristics of the external, pre-existing and legacy systems.

At *application level*, extending the information means supporting new operations and functionalities (e.g., data mining, decision-support analysis, stickers printing, etc.). Those features often result in separate applications because they are strictly related to a specific user cluster or intended for specialized demands. Nonetheless, adopting a Semantic Web approach, the design and development of those applications are simplified, because they can rely upon an (RDF) repository of integrated data, described by an ontology which formalizes their semantics and which can be re-used at application level. Besides data aggregation, the tool may also "export" (e.g. through Web Services) several general-purpose operations, which can be exploited by external applications.

At *user interface level*, extending the information means supporting new data visualization and editing. In this regard, the same (RDF) model used to represent data and the ontological formalization used to express their semantics can be exploited in order to (semi-)automatically generate the user interface. The simplest solution is the adoption of RDF Browsers, like Tabulator [15], Disco [16] or Rhizomer [17], which automatically generates an HTML presentation starting from RDF (linked) data. A more evoluted approach is represented by Semantic Web Portals [18] and by Web frameworks that exploit an ontological model to determine resources' presentation and navigational patterns (like in [19]), based on a model-driven approach in which the model is expressed by the ontology.

### 4.4 Semantics in Data Fruition

For what regards *contact organization*, an effective "semantic" approach to improve data access and fruition is *tagging*, i.e., the possibility to classify contacts along pre-defined or user-generated categories. Those classifications can then be used for searching, together with other contact-specific information.

*Contact search* can be successfully solved by adopting Semantic Web approaches:

- Formal or Directed Search implies the use of a search engine; having an ontology of the managed data, a *semantic search engine* can be employed (like, for example, Squiggle [20]). Differently from a syntactic search engine, it can support the user in disambiguating his search criteria to smoothly get to the desired contact information.
- Free or Indirected Search, on the other hand, requests for selecting, grouping and browsing through contacts on the basis of the associated information. Also in this case, the presence of an ontology enables *semantic navigation* and *faceted browsing* solutions (like BrowseRDF [21] or /facet [22]). Those approaches let the user to limit (or extend) the focus on specific contact categories, which become different as the navigation continues, always supported by the application which takes care of avoiding the user to reach a "deadlock".

For what regards the *access or reuse of contact information* in everyday PIM tools, possible solutions heavily depend on the features of different tools. We can imagine the design of plug-ins for the most commonly used tools or, whenever possible, the exploitation of importing features to migrate data to them.

### 4.5 Semantics in Privacy and Security

The privacy and security issues discussed in section 3.5 ask for the adoption and the enforcement of proper access control mechanisms, controlled by flexible security policies defined by system administrators and data owners.

This access control model should be focused on data *access rights* (read and update operations) and should be aimed at supporting the definition of *access policies*, specifying the users allowed to access or update a particular contact or kind of information. As a general rule-of-thumb, we propose to define access rights and policies in terms of *user groups* (e.g., the whole staff working on a particular project), while the object of those policies can be identified in one or more contacts or in a set of cross-over properties over all the contacts (e.g., contacts' birthday can be accessed only by the Communication department which is in charge of sending greetings messages).

The proposed access control system can exploit the common RDF model and the availability of ontologies which formalize the meaning of data. For example, policy rules could be expressed through a rule language, while specific access rights could be associated to contact "triples" through the use of Named Graphs, in addition to provenance information and update history (cf. section 4.1). Finally, data ontology can be used as a basis to express access policies, by associating specific rules to some contact classes or to their properties, therefore defining the user groups in terms of their "semantics".

Generally speaking, the contact data maintained by the tool could be seen as a (logically) centralized RDF repository, which can be protected through the use of an access control framework for RDF stores, like the ones proposed in [23] and [24]. Those frameworks support the definition of rule based policies, which control the execution of the fundamental repository operations (insert, remove and read operations) and consider also the case of triggered reasoning that can result in adding, deleting or retrieving additional information.

## 5 Conclusions and Future Work

In this paper, we presented our analysis of the needs and requirements of Contact Management, on the basis of previous experience with our GeCo prototype.

It is quite well-known that today companies cannot miss a proper and careful management of their contacts, in order to make their network grow and to cultivate a good relationship with their business partners. People clearly perceive that "knowing who you know" and finding the right contacts to start a cooperation with (e.g., finding an expert in a specific field) are becoming more and more crucial in today's business.

We explained how a Contact Management application aims at solving a real-world problem which becomes more and more apparent every day, with the birth and spreading of new systems aimed at supporting people in keeping trace of their existing contacts and to exploit the opportunity of finding new ones (like LinkedIn, for example). However, neither those social networking systems nor the widespread PIM tools or email clients we everyday use are able to give a complete answer to the general question of Contact Management. A comprehensive solution can come only if a tool succeeds in supporting not only the editing, storage and search for contacts, but also the integration with existing and running systems and processes.

Contact Management is a data-intensive field, where information semantics plays a key role in supporting the effective acquisition, management and retrieval of contacts. In this sense, we believe that the adoption of "semantic" technologies could enable more sophisticated functionalities, such as automatic contact classification, identification of potential partners and planning of a more targeted and effective communication.

Moreover, personal information about an individual is spread today in several, independent data sources: each time we establish a work relationship or register for a service, we usually have to communicate our personal information. This represents a cost, which increases when we want or need to keep this information up-to-date (e.g. when we expect to be contacted by the interlocutor). The exposed considerations lead us to believe that Contact Management could be a valuable use case for the evaluation of Semantic Web technologies.

In the next future, we are planning to extend the GeCo prototype through the progressive adoption of the semantic approaches highlighted in this paper. We would like also to move toward the use of published user profiles as a source for contacts. To this end, we could initially support the acquisition of *foaf* profiles, because of their wide adoption (over 10'000'000 *foaf* profiles on the Web in 2005), as well as the publication of a subset of the managed contacts as Linked Data.

## Acknowledgments

We wish to thank the anonymous reviewers for their useful comments. This research has been partially supported by the SEEMP EU-funded project (IST-4-027347) and by the NeP4B Italian-funded FIRB project (MIUR-2005).

## References

1. Whittaker, S., Jones, Q., Terveen, L.: Contact management: identifying contacts to support long-term communication. In: CSCW '02: Proceedings of the 2002 ACM conference on Computer supported cooperative work, New York, NY, USA, ACM Press (2002) 216–225
2. Choo, W., Detlor, B., Turnbull, D.: Information Seeking on the Web: an Integrated Model of Browsing and Searching. Fist Monday **5**(2) (2000)
3. Whittaker, S., Jones, Q., Terveen, L.: Managing long term communications: Conversation and contact management. In: HICSS '02: Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS'02)-Volume 4, Washington, DC, USA, IEEE Computer Society (2002) 115.2
4. Klyne, G., Carroll, J.J.: Resource Description Framework (RDF): Concepts and Abstract Syntax. http://www.w3.org/TR/rdf-concepts/ (2004) W3C Recommendation.
5. Carroll, J., Bizer, C., Hayes, P., Stickler, P.: Named Graphs, Provenance and Trust. In: Proceedings of the 14th International World Wide Web Conference (WWW 2005), ACM (2005) 613–622
6. Hayes, P.: RDF Semantics, section 3.3.1 Reification. http://www.w3.org/TR/rdf-mt/#Reif (2004) W3C Recommendation.

7. Boley, H., Kifer, M.: RIF Core Design. http://www.w3.org/TR/rif-core/ (2007) W3C Working Draft.
8. Prud'hommeaux, E., Seaborne, A.: SPARQL Query Language for RDF. http://www.w3.org/TR/rdf-sparql-query/ (2007) W3C Candidate Recommendation.
9. Alden, R., et al.: vCard. http://www.imc.org/pdi/vcard-21.txt (1996) A versit Consortium Specification.
10. Miller, L., Brickley, D.: The Friend of a Friend (FOAF) project. http://www.foaf-project.org/ (since 2000)
11. Bizer, C., Cyganiak, R., Heath, T.: How to Publish Linked Data on the Web. http://sites.wiwiss.fu-berlin.de/suhl/bizer/pub/LinkedDataTutorial/ (2007)
12. Adida, B., Birbeck, M.: RDFa Primer 1.0 – Embedding RDF in XHTML. http://www.w3.org/TR/xhtml-rdfa-primer/ (2007) W3C Working Draft.
13. Connolly, D., et al.: Gleaning Resource Descriptions from Dialects of Languages (GRDDL). http://www.w3.org/TR/grddl/ (2007) W3C Proposed Recommendation.
14. Kassoff, M., Petrie, C., Zen, L.M., Genesereth, M.: Semantic Email Addressing: Sending Email to People, Not Strings. In: AAAI 2006 Fall Symposium on Integrating Reasoning into Everyday Applications. (2006)
15. Berners-Lee, T., Chen, Y., Chilton, L., Connolly, D., Dhanaraj, R., Hollenbach, J., Lerer, A., Sheets, D.: Tabulator: Exploring and analyzing linked data on the semantic web. In: Proceedings of the 3rd International Semantic Web User Interaction Workshop. (2006)
16. Bizer, C., Gaub, T.: Disco - Hyperdata Browser. http://sites.wiwiss.fu-berlin.de/suhl/bizer/ng4j/disco/ (2007)
17. Garcia, R., Gil, R.: Building a semantic intraweb with rhizomer and a wiki. In: IntraWebs Workshop, 15th World Wide Web Conference, Edinburgh, UK (2006)
18. Lausen, H., Ding, Y., Stollberg, M., Fensel, D., Hernández, R.L., Han, S.K.: Semantic web portals: state-of-the-art survey. Journal of Knowledge Management **9**(5) (2005)
19. Celino, I., Della Valle, E.: Multiple vehicles for a semantic navigation across hyper-environments. In Gómez-Pérez, A., Euzenat, J., eds.: ESWC. Volume 3532 of Lecture Notes in Computer Science., Springer (2005) 423–438
20. Celino, I., Della Valle, E., Cerizza, D., Turati, A.: Squiggle: a semantic search engine for indexing and retrieval of multimedia content. In: 1st International Workshop on Semantic-enhanced Multimedia Presentation Systems, Athens, Greece (2006)
21. Oren, E., Delbru, R., Decker, S.: Extending faceted navigation for rdf data. In: ISWC. (2006)
22. Hildebrand, M., van Ossenbruggen, J., Hardman, L.: /facet: A browser for heterogeneous semantic web repositories. In Cruz, I.F., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L., eds.: International Semantic Web Conference. Volume 4273 of Lecture Notes in Computer Science., Springer (2006) 272–285
23. Reddivari, P., Finin, T., Joshi, A.: Policy based access control for a rdf store. In: Proceedings of the Policy Management for the Web Workshop. A WWW 2005 Workshop (2005)
24. Dietzold, S., Auer, S.: Access control on rdf triple stores from a semantic wiki perspective. In: Proceedings of the Scripting for the Semantic Web Workshop at the ESWC, Budva, Montenegro (2006)