Agent Oriented Data Integration

Avigdor Gal and Aviv Segev

Technion - Israel Institute of Technology,
avigal@ie.technion.ac.il, asegev@tx.technion.ac.il
Christos Tatsiopoulos, Kostas Sidiropoulos, Pantelis Georgiades
European Profiles SA
ctatsio@hol.gr

Abstract. Data integration is the process by which data from heterogeneous data sources are conceptually integrated into a single cohesive data set. In recent years agents have been increasingly used in information systems to promote performance. In this work we propose a modeling framework for agent oriented data integration to demonstrate how agents can support this process. We provide a systematic analysis of the process using real world scenarios, taken from email messages from citizens in a local government, and demonstrate two agent oriented data integration tasks, email routing and opinion analysis.

1 Introduction

Data integration is the process of combining two or more data sets together for sharing and analysis to support information management. Agents are autonomous, or semi-autonomous proactive and reactive computer software. Although there is a vast corpus of research of data integration, this research has little impact on the state-of-the-art in agent oriented systems. We believe this chasm can be attributed to the fact that most approaches rely on semantic reconciliation to be resolved first (probably manually), before attending to the more "technical" aspects of the integration. However, researchers and practitioners alike are coming to realize that there can be no solution to the delivery of integrated information unless the semantic heterogeneity problem is tackled head-on [20]. This research works towards this goal through the use of ontologies.

This approach of agent oriented data integration was recently adopted in QUALEG, a European Commission project aimed at increasing citizen participation in the democratic process.[1] In QUALEG, contexts are used to specify the input from citizens and then to provide services - routing emails to departments and performing opinion analysis on topics at the forefront of public debates.

We present the QUALEG approach towards agent oriented data integration. We first propose a modeling framework for agent oriented data integration. We then provide a systematic analysis of the process using real world scenarios, taken from email messages from citizens in a local government, and demonstrate two agent oriented data integration tasks, email routing and opinion analysis.

[1] http://www.qualeg.eupm.net/

J. Akoka et al. (Eds.): ER Workshops 2005, LNCS 3770, pp. 98–108, 2005.

2 Background and Motivation

2.1 Data Integration

Data integration has become a common theme in the information technology world. As information is increasingly becoming more complex and vast and the management of information more critical, the need to ascertain data integrity and replication is key to the reliable operation of the information system.

Many techniques are employed to promote the data integration process, such as event-based software integration [1], database schema integration [2], Web-based information integration [11], and semantic integration [3]. The field has seen the development of many tools, such as DIKE [21], Clio [17], Cupid [13], and OntoBuilder [18], to name a few.

Although there has been extensive research performed on data integration, the use of agents to promote this process has not been addressed adequately. This paper therefore presents an agent oriented approach to data integration.

2.2 Agents

In today's world, with the proliferation of computers, agents are necessary to promote the user's effective exploitation of software systems. Agents are used to initiate communications, monitor events, and perform tasks to assist users to understand the technically complex world. Agent concepts and techniques already appear in many information system architectures.

There are agents for accessing Web Information Systems (WIS) through Mobile Devices [22]. There are also multi-agent systems that assist individual investors with stock market investments [26]. Text mining agents for net actions have been also extensively analyzed [12]. Agents have been introduced into digital libraries, such as in University of Michigan Digital Library [4] and the ZUNO Digital Library (ZUNODL) [7], a commercial framework for building digital libraries. In this work we propose a framework and architecture of agents for data integration.

2.3 Context and Ontology

Contexts and ontologies are defined and used in various research areas, including philosophy, artificial intelligence, information sciences, knowledge representation, object modeling, and most recently, eCommerce applications.

Context is defined as a first class object [15]. McCarthy defines a relation $ist(\mathcal{C}, P)$, asserting that a proposition P is true in a context $\mathcal{C}$. Previous work on contexts [24] uses metadata for semantic reconciliation. It has been proposed to use a multilevel semantic network to represent knowledge within several levels of contexts [27]. This paper employs an agent based, fully automated context recognition algorithm that uses the Internet as a knowledge base and as a basis for clustering [23].

Ontology is defined as a world of systems [6]. Bunge in his seminal work provides a basic formalism for ontologies. Typically, ontologies are represented using a Description Logic [5, 8], where subsumption typifies the semantic relationship between terms; or Frame Logic [10], where a deductive inference system provides access to semi-structured data.

The realm of information science has produced an extensive body of literature and practice in ontology construction [28], ontology management [25], ontology learning [14, 9], and the use of ontology in knowledge representation source [16, 19].

This paper presents a agent oriented model for the integration of data into an ontological structure. The data structures are represented by the ontology concepts. Each ontology concept represents a possible topic or a possible opinion.

2.4 Example

To illustrate agent oriented data integration, consider the following example of the local government of Saarbrücken.

Example 1. Two ontology concepts in the ontology of Saarbrücken are:
(Perspectives du Theatre, {{⟨Öffentlichkeitsarbeit, 2⟩}, {⟨Multimedia, 1⟩},
{⟨Kulturpolitik, 1⟩}, {⟨Musik, 6⟩}, ...})
and
(Long Day School, {{⟨Förderbedarf, 1⟩}, {⟨Mathematik, 2⟩}, {⟨Musik, 2⟩},
{⟨Interkulturell, 1⟩}})
The set of descriptors define possible contexts with appropriate weights defining the importance of each descriptor in the context. There are also two ontology concepts that represent a positive opinion and a negative opinion. Each of these opinions can be ascribed to each of the above fields of interest.

The following email is received in the local government of Saarbrücken:
Eine leerer und verwaister Festivalclub, Regen und eine lustlose Band prägten das Bild der Auftaktveranstaltung des diesjährigen Festivals.

An agent can extract the following context of the email message using the algorithm in [23] (to be described later): {{⟨Musik, 8⟩} , {⟨Open Air, 1⟩}}.

An agent can map the email to the correct ontology topic which represents a field of interest and can forward the email to the correct local government representative handling this topic.

Another agent can identify the opinion of the email and store the information. This information can be statistically analyzed, integrated, and displayed as the citizens opinions on each of the fields of interest of the local government.

3 Model

Agents can be used to automatically extract context from a given text and then map context to ontology. We propose an agent oriented method for the management of data integration involved in automatic knowledge extraction and context-to-ontology mapping.

3.1 Context Recognition Algorithm

A *context* $\mathcal{C} = \left\{ \{\langle c_{ij}, w_{ij} \rangle\}_j \right\}_i$ is a set of finite set of descriptors c_{ij} from a domain $\mathcal{D}$ with appropriate weights w_{ij}, defining the importance of c_{ij}. For example, a context $\mathcal{C}$ may be a set of words (hence, $\mathcal{D}$ is a set of all possible character combinations) defining a document *Doc*, and the weights could represent the relevance of a descriptor to *Doc*. In classical Information Retrieval, $\langle c_{ij}, w_{ij} \rangle$ may represent the fact that the word c_{ij} is repeated w_{ij} times in *Doc*.

Several methods have been proposed in the literature for extracting context from text. One method proposed in the IR community is based on the principle of counting the number of appearances of each word in the text, assuming that the words with the highest number of appearances serve as the context. Variations on this simple mechanism involve methods for identifying the relevance of words to a domain and using methods such as stop-lists and inverse document frequency.

This agent oriented model employs a context recognition algorithm that uses the Internet as a knowledge base to extract multiple contexts of a given situation, based on the streaming in text format of information that represents situations [23]. This algorithm has been extensively tested and was found to obtain similar cobtexts to those proposed by human experts. This algorithm is currently part of the QUALEG solution.

The input to the algorithm is a stream, in text format, of information. The context recognition algorithm output is a set of contexts that attempts to describe the current scenario most accurately. The set of contexts is a list of words or phrases, each describing an aspect of the scenario. The context recognition algorithm consists of the following major phases: collecting data, selecting contexts for each text, ranking the contexts, and declaring the current contexts. The phase of data collection includes parsing the text and checking it against a stop-list. To improve this process, the text can be checked against a domain-specific dictionary. The result is a list of keywords obtained from the text. The selection of the current context is based on searching the Internet for relevant documents according to these keywords and on clustering the results into possible contexts. The output of the ranking stage is the current context or a set of highest ranking contexts. The set of preliminary contexts that has the top number of references, both in number of Internet pages and in number of appearances in all the texts, is declared to be the current context.

Up to this stage, the agent has achieved a set of contexts describing the given scenario. In the next stage, the agent maps these contexts to ontology concepts to achieve the automatic data integration.

3.2 Data Integration Using Contexts and Ontologies

When a context is extracted automatically from some information source (*e.g.*, an email message), the assumption is that it is correct, although it may not be extracted accurately and context descriptors may have been erroneously added or eliminated. Moreover, there may be inaccuracies in the definition of ontologies.

Therefore, to integrate the data it is necessary for the agent to map the extracted contexts to the relevant ontology concepts - a set of sets of contexts.

A context can belong to multiple context sets, which in turn can converge to different ontology concepts. Thus, one context can belong to several ontology concepts simultaneously.

For example, a context ⟨Musik, 2⟩ can be shared by many ontology concepts with interest in culture (such as schools, after school institutes, non-profit organizations, *etc.*) yet it is not in their main role definition. Such overlap of contexts in ontology concepts affects the task of email routing. The appropriate interpretation of a context of an email, when the context is part of several ontology concepts, is that the email is relevant to all such concepts. Therefore, it should be delivered to multiple departments in the local government.

A good algorithm for context extraction generates contexts in which false negatives and false positives are considered to be the exception, rather than the rule. Therefore, we would like to measure some "distance" between an extracted context and various ontology concepts, assuming a "closer" ontology concept to be better matched. To that end, we define a metric function for measuring the distance between a context and ontology concepts, as follows.

We first define distance between two weighted context descriptors $\langle c_i, w_i \rangle$ and $\langle c_j, w_j \rangle$ to be:

$$d(c_i, c_j) = \begin{cases} |w_i - w_j| & i = j \\ \max(w_i, w_j) & i \neq j \end{cases}$$

This distance function assigns greater importance to descriptors with larger weights, assuming that weights reflect the importance of a descriptor within a context. To define the best ranking concept in comparison with a given context we use Hausdorff metric. Let A and B be two contexts and a and b be descriptors in A and B, respectively. Then,

$$\begin{aligned} d(a, B) &= \inf\{d(a, b) | b \in B\} \\ d(A, B) &= \max\{\sup\{d(a, B) | a \in A\}, \sup\{d(b, A) | b \in B\}\} \end{aligned}$$

The first equation provides the value of minimal distance of an element from all elements in a set. The second equation identifies the furthest elements when comparing both sets.

Of particular interest are ontology concepts that are considered "close" under some distance metric. As an example, consider the task of opinion analysis. With opinion analysis, a system should not only judge the relevant area of interest of a given email but also determine the opinion that is expressed in it. Consider an opinion analysis task, in which opinions are partitioned into categories (*e.g.*, "for" and "against"). We can model such opinions using a common concept ontology (say, that of Perspectives du Theatre, see Example 1), with the addition of words that describe opinions. An email whose context fit with the theme of the ontology concept will be further analyzed to be correctly classified to an opinion category.

Example 2. (Email Routing) Returning to our case study example, the context $\{\{\langle \text{Musik}, 8\rangle\}, \{\langle \text{Open Air}, 1\rangle\}\}$ may be relevant to both Perspective du Theatre and Long Day School, since in both, a descriptor Musik is found, albeit with different weights. The distance between ⟨Musik, 8⟩ and ⟨Musik, 6⟩ in Perspective du Theatre is 2, and to ⟨Musik, 2⟩ in Long Day School is 6. Assume that {⟨Open Air, 1⟩} is a false positive, which does not appear in either Perspective du Theatre or in Long Day School. Therefore, its distance from each of the two points accumulation is 1 (since $\inf\{d(a, b)|b \in B\} = 1$, *e.g.*, when comparing {⟨Open Air, 1⟩} with {⟨Kulturpolitik, 1⟩}). We can therefore conclude that the distance between the context and Perspective du Theatre is 2, which is smaller than its distance from Long Day School (computed to be 6). Therefore, Perspective du Theatre will be ranked higher than Long Day School.

4 Architecture

This agent oriented method for integrating the context into the ontology concept according relevance is applied in the tasks of email routing and opinion analysis.

Email routing: The user provides QUALEG with a distance threshold t_1. Any ontology concept that matches with a context, automatically generated from an email, and its distance is lower than the threshold ($d(A, B) < t_1$) will be considered relevant, and the email will be routed accordingly.

Opinion analysis: A relevant set of ontology concepts is identified, similarly to email routing. Then for each ontology concept, the relative distance of the different opinions of that concept is evaluated. If the difference in distance is too close to call (given an additional threshold t_2), the system refrains from providing an opinion (and the email is routed accordingly). Otherwise, the email is marked with the opinion with minimal distance.

These tasks are achieved through the implementation of the agent oriented Qualeg architecture, which consists of the following main seven components: (1) Agora - A Web interface to the system through which the citizen interacts via emails, chats and forums with the civil servant. (2) Datamart - The component that stores all the Qualeg data. (3) Qualeg ontology - A multilingual ontology describing the public and e-government issues. (4) Knowledge Extractor - The previously described context extraction algorithm used by the agents. (5) Qualeg Workflow - The component that handles the flow of processes relevant to the public servants and administrations. (6) A set of Intelligent Agents, which in the backstage handle the main control of the Qualeg system, acting asynchronously and handling the data to be communicated among various modules and passing control this way. (7) A set of Web Services offered for seamless data handling to and from the Datamart.

There are five different agents in the system, classified according to their task as follows: Knowledge Extraction, Opinion Analysis, Off-line Questionnaires, Email, and Email Handler.

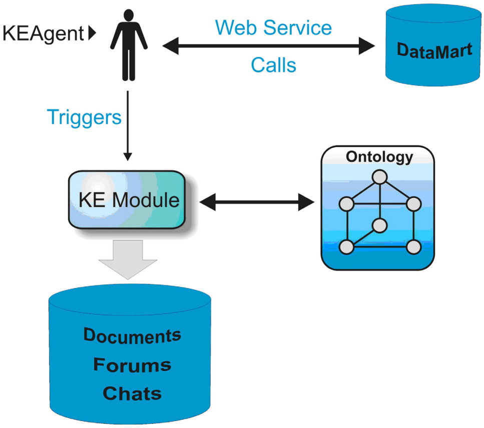

Fig. 1. Agent Architecture

The main focus of this part is on the use of data integration intelligent agent interactions with the rest of the Qualeg modules as a means for both asynchronous and synchronous control. The intelligent agents have been developed in the JADE platform and in line with the FIPA specifications for interoperable intelligent multi-agent systems. The following agents are provided in the Qualeg Architecture solution:

Knowledge Extraction Agent. The Knowledge Extraction Agent (KE Agent) illustrated in Figure 1 has the responsibility to trigger the Knowledge Extraction Module so that the context of the stored information is regularly analyzed. There are four types of documents that should be analyzed: documents uploaded to AGORA, text in forums, chats, and incoming e-mail messages. In particular, the KE Agent performs periodical searches in the platform's databases for new information to be analyzed. Every transaction with the database is carried out by means of Web services. If new documents are found, the agent triggers the previously described knowledge extraction algorithm on them. Hence, the KE Agent parses all the required information - such as document id, document name, document url - to the KE module. The KE module performs the mapping with reference to an ontology, which defines the set of concepts and their relationships. After the KE process is completed, a set of keywords is stored in a database.

Opinion Analysis Agent. Similarly to the KE Agent, the Opinion Analysis Agent (OAAgent) regularly searches in QUALEG's databases to find which documents have to be analyzed by the Opinion Analysis Module (OA Module). Once again, all the agent's database transactions are carried out through Web service calls. If documents requiring analysis are found, the agent triggers the opinion analysis algorithm on them in the same way as the KE agent. Opinion Analysis output is an ontology concept and a list of words.

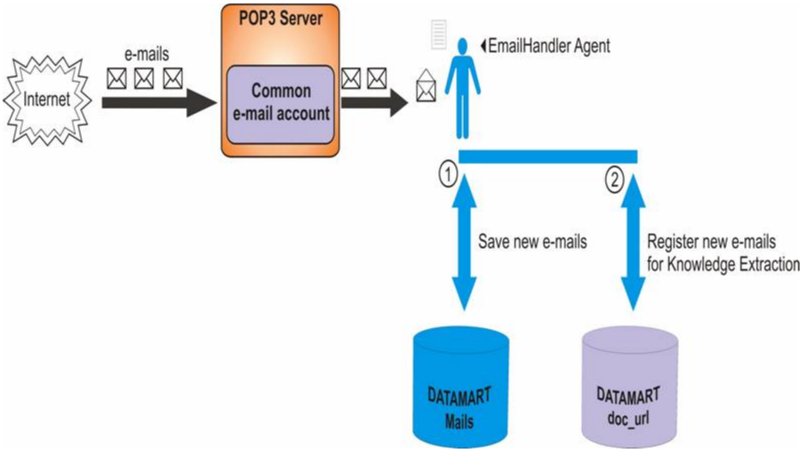

Fig. 2. Agent System Architecture

Email Handler Agent. All incoming e-mails that have been sent to a common e-mail account of the local government are gathered by an agent. The EmailHandlerAgent illustrated in Figure 2 disassembles each e-mail into its parts and distils the contained text. Next, the agent registers the new e-mail to a designated database in DATAMART. In particular, the EmailHandlerAgent, using Web services, saves information concerning the e-mail, *eg. sender, subject, body etc.*

5 Results

The aim of the QUALEG project is to support the electronic interactions between civil servants and citizens. Our experiment domain was the Perspectives Festival of May 15-21, 2005 in Saarbrücken (http://www.perspectives-sb.de/) along with similar data from the previous year's festival, which included films, theatre, street events, music, etc. Given the daily communications (in German) about this event, which consisted primarily of emails from citizens to the city hall or press releases and announcements from the city outward, our challenge was to analyze this material and provide a useful set of classifications so that the materials could be rapidly understood and sent to the appropriate people for response. Two different agent systems were developed, separating the task of knowledge extraction from that of opinion analysis. The main difference between the two agents is that the knowledge extraction agent avoids the language specific implementation and bases its analysis techniques on the use of a large corpus of relevant documents taken from the Internet, while the opinion analysis agent uses techniques from IR and NLP to improve content understanding. The systems analyzed the materials by topic (ticket/travel information, finances, organization,

Table 1. Context Recognition / Knowledge Extraction Agent

Precision	85.37 %
Recall	84.34 %
F-Score	84.85 %

etc.) and opinion (positive, negative, etc.). The system's average performance achieved high correspondence to human results for the different classes.

Our first experiment included 104 different emails to analyze the knowledge extraction agent. Table 1 summarizes the comparison of the results of the context recognition knowledge extraction agent to the human judgements. Our second experiment included 72 different emails to analyze the opinion analysis agent. Table 2 summarizes a similar comparison of the results of the opinion analysis agent. Both tables contain the precision, recall, and the weighted toward Precision F-score obtained. These results show the promising ability of our agents to integrate the data from the citizens with the local government specifications.

Table 2. Opinion Analysis Agent

Precision	78.95 %
Recall	69.23 %
F-Score	73.77 %

6 Conclusion

Data integration is a key field in the management of information systems today. The use of autonomous or semi-autonomous agents can effectively promote the process of data integration. The paper presents a modeling framework for agent oriented data integration to demonstrate how agents can support this process. The agent architecture and the analysis of the empirical results are based on real life scenarios, taken from email messages from citizens in a local government, and demonstrate two agent oriented data integration tasks, email routing and opinion analysis.

Initial tests show that the algorithm also achieves high performance compared to manually integrated data. Future directions of research include automatic agent responses to incoming data based on the previously integrated data.

Acknowledgments

The work of Gal was partially supported by two European Commission 6^{th} Framework IST projects, QUALEG and TerreGov, and the Fund for the Promotion of Research at the Technion.

References

1. D. Barret, L. Clarke, P. Tarr, and A. Wise. A framework for event-based software integration. *ACM Transactions on Software Engineering and Methodology*, 5(4), 1996.
2. C. Batini, M. Lenzerini, and S. Navathe. A comparative analysis of methodologies for database schema integration. *ACM Computing Surveys*, 18(4):323–364, December 1986.
3. S. Bergamaschi, S. Castano, M. Vincini, and D. Beneventano. Semantic integration of heterogeneous information sources. *Data & Knowledge Engineering*, 36(3), 2001.
4. W. P. Birmingham, E. H. Durfee, T. Mullen, and M. P. Wellman. The distributed agent architecture of the university of michigan digital library. In *AAAI Spring Symposium on Information Gathering*, 1995.
5. A. Borgida and R. J. Brachman. Loading data into description reasoners. In *Proceedings of the 1993 ACM SIGMOD international conference on Management of data*, pages 217–226, 1993.
6. M. Bunge. *Treatise on Basic Philosophy: Vol. 4: Ontology II: A World of Systems.* D. Reidel Publishing Co., Inc., New York, NY, 1979.
7. D. Derbyshire, I. A. Ferguson, J. P. Muller, M. Pischel, and M.Wooldridge. Agent-based digital libraries: Driving the information economy. In *Sixth IEEE Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*, 1997.
8. F.M. Donini, M. Lenzerini, D. Nardi, and A. Schaerf. Reasoning in description logic. In G. Brewka, editor, *Principles on Knowledge Representation, Studies in Logic, Languages and Information*, pages 193–238. CSLI Publications, 1996.
9. V. Kashyap, C. Ramakrishnan, C. Thomas, and A. Sheth. Taxaminer: An experimentation framework for automated taxonomy bootstrapping. *International Journal of Web and Grid Services, Special Issue on Semantic Web and Mining Reasoning*, September 2005. to appear.
10. M. Kifer, G. Lausen, and J. Wu. Logical foundation of object-oriented and frame-based languages. *Journal of the ACM*, 42, 1995.
11. C.A. Knoblock, S. Minton, J.L. Ambite, N. Ashish, I. Muslea, A. Philpot, and S. Tejada. The Ariadne approach to web-based information integration. *International Journal of Cooperative Information Systems (IJCIS)*, 10(1-2):145–169, 2001.
12. Y. Kusumura, Y. Hijikata, and S. Nishida. Text mining agent for net auction. In *ACM Symposium on Applied Computing*, 2004.
13. J. Madhavan, P.A. Bernstein, and E. Rahm. Generic schema matching with Cupid. In *Proceedings of the International conference on very Large Data Bases (VLDB)*, pages 49–58, Rome, Italy, September 2001.
14. A. Maedche and S. Staab. Ontology learning for the semantic web. *IEEE Intelligent Systems*, 16, 2001.
15. J. McCarthy. Notes on formalizing context. *In Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, 1993.
16. D.L. McGuinness, R. Fikes, J. Rice, and S. Wilder. An environment for merging and testing large ontologies. In *Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR2000)*, 2000.
17. R.J. Miller, M.A. Hernàndez, L.M. Haas, L.-L. Yan, C.T.H. Ho, R. Fagin, and L. Popa. The Clio project: Managing heterogeneity. *SIGMOD Record*, 30(1):78–83, 2001.

18. G. Modica, A. Gal, and H. Jamil. The use of machine-generated ontologies in dynamic information seeking. In C. Batini, F. Giunchiglia, P. Giorgini, and M. Mecella, editors, *Cooperative Information Systems, 9th International Conference, CoopIS 2001, Trento, Italy, September 5-7, 2001, Proceedings*, volume 2172 of *Lecture Notes in Computer Science*, pages 433–448. Springer, 2001.
19. Fridman N. Noy and M.A. Musen. PROMPT: Algorithm and tool for automated ontology merging and alignment. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-2000)*, pages 450–455, Austin, TX, 2000.
20. A.M. Ouksel and A.P. Sheth. Semantic interoperability in global information systems: A brief introduction to the research area and the special section. *SIGMOD Record*, 28(1):5–12, March 1999.
21. L. Palopoli, L.G. Terracina, and D. Ursino. The system DIKE: Towards the semi-automatic synthesis of cooperative information systems and data warehouses. In *PADBIS-DASFAA*, pages 108–117, 2000.
22. A. C. Ramos, J. Gensel, M. Villanova-Oliver, and H. Martin. Adapted information retrieval in web information systems using pumas. In *Workshop on Agent-Oriented Information Systems (AOIS)*, 2005.
23. A. Segev. Identifying the multiple contexts of a situation. In *Proceedings of IJCAI-Workshop Modeling and Retrieval of Context (MRC2005)*, 2005.
24. M. Siegel and S. E. Madnick. A metadata approach to resolving semantic conflicts. In *Proceedings of the 17th International Conference on Very Large Data Bases*, pages 133–145, 1991.
25. P. Spyns, R. Meersman, and M. Jarrar. Data modelling versus ontology engineering. *ACM SIGMOD Record*, 31(4), 2002.
26. S. C. Sundararajan, S. Sankarlal, and A. Kumar. Inca (investor network collaborative architecture) a method in the madness of wall street. In *Workshop on Agent-Oriented Information Systems (AOIS)*, 2005.
27. V. Terziyan and S. Puuronen. Reasoning with multilevel contexts in semantic metanetwork. In R. Nossun P. Bonzon, M. Cavalcanti, editor, *Formal Aspects in Context*, pages 107–126. Kluwer Academic Publishers, 2000.
28. B.C. Vickery. *Faceted classification schemes.* Graduate School of Library Service, Rutgers, the State University, New Brunswick, N.J., 1966.