Some DLV Applications for Knowledge Management

{leone,ricca}@mat.unical.it
[2] Exeura Srl, Via Pedro Alvares Cabrai - C.da Lecco 87036 Rende (CS), Italy
salvatore.iiritano@exeura.com

Abstract. Even if the industrial exploitation of the DLV system has started very recently, DLV already has a history of applications on the industrial level. The most valuable applications from a commercial viewpoint are those in the area of Knowledge Management. They have been realized by the company EXEURA s.r.l. - a spin-off company of the University of Calabria having a branch also in Chicago - with the support of the DLVSYSTEM s.r.l.. DLV applications in this area have not been realized directly, but through some specializations of DLV into Knowledge Management (KM) products for Text Classification, Information Extraction, and Ontology Representation and Reasoning. After briefly describing these KM products, we report on their recently-released successful applications.

1 Introduction

Answer Set Programming (ASP) [2,3] is a powerful logic programming language. In its general form, allowing for disjunction in rule heads [1] and nonmonotonic negation in rule bodies, ASP can represent *every* problem in the complexity class Σ_2^{P} and Π_2^{P} (under brave and cautious reasoning, respectively) [4]. The high knowledge modeling power of ASP, and the availability of efficient ASP systems, has implied a renewed interest in this formalism in recent years, due to the need for representing and manipulating complex knowledge, arising in Artificial Intelligence as well as in other emerging areas, like Knowledge Management and Information Integration.

One of the most relevant ASP systems is DLV [13]. The DLV system is the product of more than twelve years of research and development and is the state-of-the-art implementation of disjunctive ASP. DLV is widely used by researchers all over the world, and it is competitive, also from the viewpoint of efficiency, with the most advanced ASP systems. Indeed, at the First Answer Set Programming System Competition [5] DLV won in the Disjunctive Logic Programming category; and DLV finished first also in the general category MGS (Modeling, Grounding, Solving — also called royal competition, which is open to all ASP systems). Importantly, DLV is profitably employed in many real-word applications, and has stimulated quite some interest also in industry. In the following, we report on the most valuable applications of DLV, which are industrially developed by the company EXEURA s.r.l., a spin-off company of the University of Calabria, with the support of the DLVSYSTEM s.r.l., another spin-off company maintaining the system. Actually, many DLV applications have not been realized directly, but through some specializations of the system in products for Text Classification, Information Extraction, and Ontology Representation and Reasoning.

E. Erdem, F. Lin, and T. Schaub (Eds.): LPNMR 2009, LNCS 5753, pp. 591–597, 2009.

In this paper, we first describe the DLV-based Knowledge Management products, we then overview some successful applications of these products, and we briefly report also on some further applications exploiting DLV directly. The described applications fall in many different domains, including Team Building in a Seaport, E-Tourism, E-Government, etc..

2 DLV-Based Commercial Systems

The three main industrial products of Exeura s.r.l. that are strongly based on the DLV system are, namely: OntoDLV [6,7], OLEX [8,9], HiLeX [10,11]. OntoDLV is an ontology management and reasoning system; OLEX is a document classification system; and, HiLeX is an information extraction system. In the following, we provide a brief description of the main features of those systems.

OntoDLV. Traditional ASP in not well-suited for ontology specifications, since it does not directly support features like classes, taxonomies, individuals, etc. Moreover, ASP systems are a long way from comfortably enabling the development of industry-level applications, mainly because they lack important tools for supporting programmers. Both the above-mentioned issues were addressed in OntoDLV [6,7] a system for ontologies specification and reasoning. Indeed, OntoDLV implements a powerful logic-based ontology representation language, called OntoDLP, which is an extension of (disjunctive) ASP with all the main ontology constructs including classes, inheritance, relations, and axioms. OntoDLP is strongly typed, and includes also complex type constructors, like lists and sets. Importantly, OntoDLV supports a powerful interoperability mechanism with OWL, allowing the user to retrieve information from external OWL Ontologies and to exploit this data in OntoDLP ontologies and queries.

Using OntoDLV, domain experts can create, modify, store, navigate, and query ontologies thanks to a user-friendly visual environment; at the same time, application developers can easily implement knowledge-intensive applications embedding OntoDLP specifications using a complete Application Programming Interface (API) [12]. Moreover, OntoDLV facilitates the development of complex applications in a user-friendly visual environment; it is endowed with a robust persistency-layer for saving information transparently on a DBMS, and it seamlessly integrates the DLV system [13].

OLEX. The OntoLog Enterprise Categorizer System (OLEX) [8,9] is a corporate classification system supporting the entire content classification life-cycle, including document storage and organization, ontology construction, pre-processing and classification. OLEX exploits a reasoning-based approach to text classification which synergically combines: (i) ontologies for the formal representation of the domain knowledge; (ii) pre-processing technologies for a symbolic representation of texts and (iii) ASP as categorization rule language. Logic rules, indeed, provides a natural and powerful way to encode how document contents may relate to ontology concepts.

More in detail, the main task of OLEX is text categorization, which is is the task of assigning documents to predefined categories on the basis of their content. To this end, in the system, ontologies are exploited for modeling the domain knowledge; and, with each concept of a given ontology is associated a specific ASP program (containing

the *classification rules*) that is used to recognize concepts in a text. Classification rules can be either manually specified or automatically determined [9]. Clearly, the system has to pre-process the input documents in order to produce a logic representation of their content. The OLEX pre-processor performs the following tasks: Pre-Analysis and Linguistic Analysis. The former consists of document normalization, structural analysis and tokenization; whereas the latter includes lexical analysis, which determines the Part of Speech (PoS) of each token, reduction (elimination of the stop words), and frequency analysis. The output of the pre-processing phase for a document is a set of facts modeling its content. The obtained facts are then fed into the DLV system together with the classification rules to compute an association between the processed document and ontology concepts.

H$\imath$LεX. [10,11] is an advanced system for ontology-based information extraction from semi-structured and unstructured documents. In practice, HiLeX implements a semantic approach to the information extraction problem by exploiting: (i) ontologies as knowledge representation formalism; (ii) a general document representation model able to unify different document formats (html, pdf, doc, ...); and, (iii) the definition of a formal attribute grammar able to describe, by means of declarative rules, objects/classes w.r.t. a given ontology.

HiLeX is based on OntoDLP for describing ontologies, since this language perfectly fits the definition of semantic extraction rules. Regarding the unified document representation, the idea is that a document (unstructured or semi-structured) can be seen as a suitable arrangement of objects in a two-dimensional space. Each object has its own semantics, is characterized by some attributes and is located in a two-dimensional area of the document called *portion*. A portion is defined as a rectangular area univocally identified by four cartesian coordinates of two opposite vertices. Each portion "contains" one or more objects and an object can be recognized in different portions.

The language of HiLeX is founded on the concept of *ontology descriptor*. A "descriptor" looks like a production rule in a formal attribute grammar, where syntactic items are replaced by ontology elements, and where extensions for managing two-dimensional objects are added. Each descriptor allows us to describe: (i) an ontology object in order to recognize it in a document; or (ii) how to "generate" a new object that, in turn, may be added in the original ontology. Note that an object may also have more than one descriptor, thus allowing one to recognize the same kind of information when it is presented in different ways. It is worth noting that, most of the existing information extraction approaches do not work in a semantical way and they are not independent of the specific type of document they process. On the contrary, the approach implemented in HiLeX allows for recognizing, extracting and structuring relevant information form heterogeneous sources.

3 Some Commercial Applications

In this Section, we report a brief description of a number of applications developed by Exeura s.r.l. which employ the commercial products based on DLV.

Team Building in the Gioia-Tauro Seaport. The port authority of Gioia Tauro is employing a system, based on OntoDLV, for the automatic generation of the teams of employees. The problem here is to produce an optimal allocation of the available personnel of the international seaport of Gioia Tauro in such a way that the right processing of the shoring cargo boats is guaranteed at the minimum cost. To this end several constraints have to be satisfied, concerning the size and the slot occupied by cargo boats, the allocation of each employee (e.g. each employee might be employed in several roles of different responsibility, roles have to be played by the available units by possibly applying a round-robin policy, etc.), etc.. The system can build new teams or complete the allocation automatically when the roles of some key employees are fixed manually.

In this application, the domain is modeled by exploiting OntoDLV, and a set of suitably defined reasoning modules is exploited for finding the desired allocation. In this application, the pure declarative nature of the language allowed for refining and tuning both problem specifications and encodings together while interacting with the stakeholders of the seaport. It is worth noting that, the possibility of modifying (by editing text files) in a few minutes a complex reasoning task (e.g. by adding new constraints), and testing it "on-site" together with the customer is a great advantage of our approach.

E-Tourism. IDUM is an e-tourism system developed in the context of the project "IDUM: Internet Diventa Umana" funded by the administration of the Calabria Region. The IDUM system helps both employees and customers of a travel agency in finding the best possible travel solution in a short time. It can be seen as a "mediator' system finding the best match between the offers of the tour operators and the requests of the turists. More in detail, in the IDUM system, behind the web-based user interface, there is an intelligent core that exploits an OntoDLV ontology for both modeling the domain of discourse (i.e., geographic information, user preferences, and touristic offers, etc.) and storing the available data. The ontology is automatically populated by extracting the information contained in the touristic leaflets produced by tour operators and received by the travel agency attached to email messages. It is worth noting that, the received e-mails are human-readable, and the details are often contained in email-attachments of different format (plain text, pdf, gif, or jpeg files) and structure that might contain a mix of text and images. The HɿLɛX system allows for automatically processing the received contents, and to populate the ontology with the data extracted from touristic leaflets. Once the information is loaded on the ontology, the user can perform an intelligent search for selecting the holiday packages that best fit his needs. Basically, IDUM tries to mimic the behavior of the typical employee of a travel agency by running a set of specifically devised logic programs that reason on the information contained in the ontology. The result is a system that is able to search in a huge database of automatically classified offers. IDUM combines the speed of computers with the knowledge of a travel agent.

Automatic Itinerary Search. In this application, a web portal conceived for better exploiting the whole transportation system, including both public and private companies, of the Italian region Calabria. The user can ask for the automatic construction of a complete itinerary from a given place to another in the region, and the system provides it with several possible solutions depending on both the available resources and user-selected options (e.g., preferred mean, preferred transportation company, minimization

of travel distances and/or travel times etc.) The system is very precise, it tells you where and what time to catch your bus/train, where to get off and transfer, how long your trip will take, walking directions etc. This service was implemented by exploiting an OntoDLV ontology that models all the available transportation means, their timetables, and a map with all the streets, bus stops, railways and train stations etc. A set of specifically devises ASP programs are used to build the required itineraries.

e-Government. In this field, an application of the OLEX system was developed, in which legal acts and decrees issued by public authorities are classified. The system employes an ontology based on both TE.SE.O (Thesaurus of the Senato della Repubblica Italiana), an archive that contains a complete listing of words arranged in groups of synonyms and related concepts regarding juridical terminology employed by the Italian Parliament, and a set of categories identified by analyzing a corpus of 16,000 documents of the public administration. The system was validated with the help of the employees of the Calabrian Region administration, and it performed very well by obtaining an f-measure of 92% and a mean precision of 96% in real-world documents.

e-Medicine. OLEX was employed for developing a system able to classify automatically case histories and documents containing clinical diagnoses. The system was commissioned, with the goal of conducting epidemiological analyses, by the ULSS n.8 (which is, a local authority for health services) of the area of Asolo, in the Italian region Veneto. Basically, available case histories are classified by the system in order to help the analysts of the ULSS while browsing and searching documents regarding specific pathologies, supplied services, or patients living in a given place etc. The application exploits an ontology of clinical case histories based on both the MESH (Medical Subject Headings) ontology and ICD9-CM a system employed by the Italian Ministry of the Heath for handling data regarding medical services (e.g. X-Rays analyses, plaster casts, etc.). The analyzed documents are stored in PDF documents and contain medical reports, hospital discharge forms, clinical analysis results etc. Classification rules were manually devised and taken into account, beside the extracted linguistic information, also the metadata contained in the case history forms. The system has been deployed and is currently employed by the personnel of the ULSS of Asolo.

4 Other Applications

The European Commission funded a project on Information Integration, which produced a sophisticated and efficient data integration system, called INFOMIX, which uses DLV at its computational core [14]. The powerful mechanisms for database interoperability, together with magic sets [15,16] and other database optimization techniques, which are implemented in DLV, make DLV very well-suited for handling information integration tasks. And DLV (in INFOMIX) was succesfully employed to develop in a real-life integration system for the information system of the University of Rome “La Sapienza” The DLV system has been experimented also with an application for Census Data Repair [17], in which errors in census data are identified and eventually repaired. DLV has been employed at CERN, the European Laboratory for Particle

Physics, for an advanced deductive database application that involves complex knowledge manipulation on large-sized databases. The Polish company Rodan Systems S.A. has exploited DLV in a tool for the detection of price manipulations and unauthorized use of confidential information, which is used by the Polish Securities and Exchange Commission. In the area of self-healing Web Services, moreover, DLV is exploited for implementing the computation of minimum cardinality diagnoses [18].

References

1. Minker, J.: On Indefinite Data Bases and the Closed World Assumption. In: Loveland, D.W. (ed.) CADE 1982. LNCS, vol. 138, pp. 292–308. Springer, Heidelberg (1982)
2. Gelfond, M., Lifschitz, V.: The Stable Model Semantics for Logic Programming. In: Proc. of ICLP, pp. 1070–1080. MIT Press, Cambridge (1988)
3. Gelfond, M., Lifschitz, V.: Classical Negation in Logic Programs and Disjunctive Databases. New Generation Computing 9, 365–385 (1991)
4. Eiter, T., Gottlob, G., Mannila, H.: Disjunctive Datalog. ACM Transactions on Database Systems 22(3), 364–418 (1997)
5. Gebser, M., Liu, L., Namasivayam, G., Neumann, A., Schaub, T., Truszczyński, M.: The first answer set programming system competition. In: Baral, C., Brewka, G., Schlipf, J. (eds.) LPNMR 2007. LNCS (LNAI), vol. 4483, pp. 3–17. Springer, Heidelberg (2007)
6. Ricca, F., Gallucci, L., Schindlauer, R., Dell'Armi, T., Grasso, G., Leone, N.: OntoDLV: an ASP-based system for enterprise ontologies. Journal of Logic and Computation (2009)
7. Ricca, F., Leone, N.: Disjunctive Logic Programming with types and objects: The DLV$^+$ System. Journal of Applied Logics 5(3), 545–573 (2007)
8. Cumbo, C., Iiritano, S., Rullo, P.: OLEX – A Reasoning-Based Text Classifier. In: Alferes, J.J., Leite, J. (eds.) JELIA 2004. LNCS (LNAI), vol. 3229, pp. 722–725. Springer, Heidelberg (2004)
9. Rullo, P., Policicchio, V.L., Cumbo, C., Iiritano, S.: Effective Rule Learning for Text Categorization. IEEE Transactions on Knowledge and Data Engineering - TKDE-2007-07-0386.R3
10. Ruffolo, M., Manna, M.: HiLeX: A System for Semantic Information Extraction from Web Documents. In: ICEIS (Selected Papers). LNCS(LNBIP), vol. 3, pp. 194–209. Springer, Heidelberg (2008)
11. Ruffolo, M., Leone, N., Manna, M., Saccà, D., Zavatto, A.: Exploiting ASP for Semantic Information Extraction. In: Proc. of ASP 2005, Bath, UK, July 2005, pp. 248–262 (2005)
12. Gallucci, L., Ricca, F.: Visual Querying and Application Programming Interface for an ASP-based Ontology Language. In: Vos, M.D., Schaub, T. (eds.) Proc. of SEA 2007, pp. 56–70 (2007)
13. Leone, N., Pfeifer, G., Faber, W., Eiter, T., Gottlob, G., Perri, S., Scarcello, F.: The DLV System for Knowledge Representation and Reasoning. ACM TOCL 7(3), 499–562 (2006)
14. Leone, N., Gottlob, G., Rosati, R., Eiter, T., Faber, W., Fink, M., Greco, G., Ianni, G., Kałka, E., Lembo, D., Lenzerini, M., Lio, V., Nowicki, B., Ruzzi, M., Staniszkis, W., Terracina, G.: The INFOMIX System for Advanced Integration of Incomplete and Inconsistent Data. In: Proc. of ACM SIGMOD (SIGMOD 2005), Baltimore, USA, pp. 915–917. ACM Press, New York (2005)
15. Cumbo, C., Faber, W., Greco, G., Leone, N.: Enhancing the magic-set method for disjunctive datalog programs. In: Demoen, B., Lifschitz, V. (eds.) ICLP 2004. LNCS, vol. 3132, pp. 371–385. Springer, Heidelberg (2004)
16. Faber, W., Greco, G., Leone, N.: Magic Sets and their Application to Data Integration. Journal of Computer and System Sciences 73(4), 584–609 (2007)

17. Franconi, E., Palma, A.L., Leone, N., Perri, S., Scarcello, F.: Census Data Repair: A Challenging Application of Disjunctive Logic Programming. In: Nieuwenhuis, R., Voronkov, A. (eds.) LPAR 2001. LNCS (LNAI), vol. 2250, pp. 561–578. Springer, Heidelberg (2001)
18. Friedrich, G., Ivanchenko, V.: Diagnosis from first principles for workflow executions. TR, Alpen Adria University, Klagenfurt, Austria (2008), http://proserver3-iwas.uni-klu.ac.at/download_area/Technical-Reports/technical_report_2008_02.pdf