Towards an Intelligent and Deeply Automated Shopping Bot

M. Eddahibi
IRF-SIC Laboratory,ENCG
Ibn Zohr University
Agadir, Morocco

A. Nejeoui
SilPro Laboratory, National School of Applied sciences, Cadi Ayyad University
Marrakesh, Morocco

C. Cherkaoui
IRF-SIC Laboratory, ENCG
Ibn Zohr University
Agadir, Morocco

ABSTRACT

Price comparison is an increasingly common usage of the internet. Its goal is to locate access and compare relevant information for specific products from retailers from whom users can buy. This work is devoted to the design and the development of a shopping bot with the aim of overcoming the well-known difficulties in price comparison area. iShopBot is a shopping bot that combines several technologies, as semantic web, NLP, Multi-Agent systems and web data mining.

General Terms

Shopping decision aids.

Keywords

Shopping bot, Semantic Web, Automated categorization, Semantic search, NLP, Multi-agents system, AML, Information retrieval.

1. INTRODUCTION

Shopping bots are software agents for decision support. They make it easy for customers who need to find the best offers from e-vendors for specific products. They thus play the role of broker. Their use has become a more common activity and a foundation pillar in e-commerce field.

Three methods are used for product search in shopping bots. The first method is based on navigating through product categories. The user selects a root category and recursively select a sub-category of the current category until the last category. This last category haven’t any sub-category, it contain only products. The second method is based on an internal search engine that answers queries about the name of a specific product. The third method combine the two previous methods and use the internal search engine to find a category of products, then the user have to navigate among the categories tree to find the targeted product. This research aims to improve shopping bots performance, by optimizing results accuracy and minimizing search time and effort in the three cited search methods. In fact, for an efficient navigation search the shopping bot must establish a consistent products categorization. To have relevant result in query based method, using the internal search engine, the shopping bot should take benefit from semantic web [1] tools. Indeed, these tools will enable the search engine to better understand user’s queries. For the third search method, evidently, the shopping bot must combine both a consistent categorization and the use of semantic web tools.

Once the target product is found, the shopping bot displays a list of e-vendors sorted using criteria. The criteria concerns in most cases price value, guarantee duration and price, delivery time and fees. To be a good online purchase-decision aid tool, a shopping bot must provide the user with up-to-date data. Two methods are used to collect e-vendors product data:

- By crawling e-vendors web sites and extracting the useful information
- Using web-services or e-vendors product data files.

In this paper we present a new intelligent shopping bot (iShopBot), based on multi-agent system, that address most important issues that hinder widespread shopping bots efficiency namely, accuracy, availability and update frequency for crawling-based extracted data. iShopBot provides a sophisticated users queries interpretation system that besides the traditional search mechanism, offers an intelligent question answering system. iShopBot provides users with a set of relevant information about the searched products. This information set is collected from manufacturers’ websites and discussion forums. The search and processing of products information is a tedious task because of distributed nature of e-vendors, the big amount and random available information with unpredictable update time. Therefore, MAS technology based system is an unavoidable solution for developing such shopping bot.

2. RELATED WORKS

The first shopping bot was BargainFinder [2] used for the comparison of music CDs. It was developed in the 90 years of the previous century. Since, several other shopping bot were developed. But until now and to the author's knowledge, fewer are the shopping bots that use deeply web semantic and Multi-agent system technologies in their operating mechanism. WebShopper+ [3] is an example of projects realized in this area. It is a cross-language comparison-shopping agent which is used to perform online price comparisons of computer books. WebShopper+ is based on a semi-automatic products categorization and it uses a restricted range of products (computer books). However, it deals with the very important issue of cross-language comparison, delivery fees and exchange rates.

3. PRODUCTS EXTERNAL WEB DATA COLLECTION

The World Wide Web is rich with information about all kind of commercial products. Actually, the customer is interested by various type of information concerning specific products. In fact, the comparison process is more complex than that provided by all existing shopping bots. Price comparison is an advanced step in the shopping process which begins with awareness step and finish with sale step. Several intermediary steps have a great effect on customers’ product choice. Indeed, customers need not only to compare e-vendors prices but before they also need to compare several things as brands. Therefore, besides search over brands websites, a significant number of clients are using social media as a resource when shopping. Products information is available on social networks [4], because customers have the ability to broadcast their recommendations and their opinions over these networks. In order to address this issue iShopBot provides its users with categorized search results that concern each product brand, product information on social networks and brand customer service.

4. ONTOLOGY CREATION

The main goal of the Semantic Web is to extend the current web, from human-readable to machine-understandable, by encoding some of the semantics of resources in a machine processable way. Computers will be able to search, process and present the content of these resources. Ontologies are one of the pillars of the Semantic Web that are used structuring knowledge. Ontologies provide a modeling of specific domain by defining concepts and relations among them.

Ontologies have gained much importance in several fields, such as natural language processing, information retrieval, and knowledge representation. Ontologies can be constructed in three different ways:

- Manual
- Semi-automatic
- Automatic

The present work uses an automatic method for ontology learning and population [5].

4.1 Ontology architecture

Three different approaches [6] can be used for ontology creation according to their architectures:

- Single ontology architecture where a global ontology provides a vocabulary for the specification of the semantics. This vocabulary is shared between all information sources.
- Multiple ontology architecture where each information source is described by its own ontology
- Hybrid ontology architecture is similar to multiple ontology architecture where each source is described by its own ontology. In addition to the multiple ontology a global ontology that contains a shared vocabulary is used.

The hybrid ontology architecture is the most adequate architecture for the construction of the present domain ontology because it supports heterogeneous views. Indeed, single global ontology that combines different product catalogues is very difficult. This is the case of e-vendors websites provide product specifications but they refer to heterogeneous product catalogues which categorize the products. In the multiple ontology architecture the absence of a shared vocabulary makes it extremely difficult to compare different e-vendors products' ontologies.

The hybrid ontology architecture is simple because all e-vendors products' ontologies use common vocabulary while being based on several heterogeneous information sources.

The local ontologies that compose the multiple ontology part of the hybrid ontology (fig.1) in the present work are learned from each e-vendor website.

The global part of the hybrid ontology is enriched from Internet by using resulting multiple ontologies terms as seeds for information crawling. Enrichment consists in identifying concepts, terms and relationships, and then integrating them in the existing ontology.

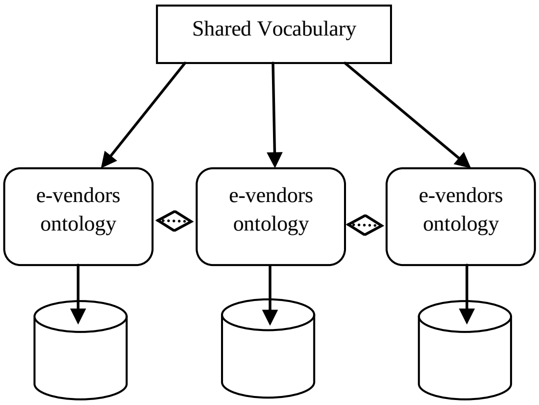

Fig 1: iShopBot ontology architecture

4.2 Ontology learning and population

There are several methodologies that address the issue of the learning and population of ontologies. iShopBot is based on Text2Onto [7] that uses a Probabilistic Ontology Model. This model indicates the degree of certainty of algorithm results. Certainty degrees are used to provide users with relevant instances filtering. Text2Onto uses a set of algorithms to calculate the probability for each modeling primitive:

To measure Concepts relevance of a certain term with respect to a corpus Text2Onto calculates Relative Term Frequency (RTF), TFIDF (Term Frequency Inverted Document Frequency), Entropy and the C-value/NC-value method.

Subclass-of Relations are learned using various algorithms that exploits the hypernym structure of WordNet [8] and matches Hearst patterns in the corpus as well as in the World Wide Web. It also applies linguistic heuristics mentioned.

JAPE expressions matching some patterns are used to discover mereological (part-of) relations in the corpus. Then an algorithm counts occurrences of patterns indicating a part-of relation between two terms.

In order to learn general relations, a shallow parsing strategy is used to extract subcategorization frames accompanied with frequency of the terms.

To find instance-of relations a similarity-based approach extracting context vectors is used. It assigns instances to the concept corresponding to the vector similar to their own vector.

Concepts equivalence is measured by algorithms calculating terms similarity on the basis of contextual features extracted from the corpus.

4.3 Products categorization

A clearly defined products e-catalog is a foundation pillar for an e-business system. Creating e-catalog is the based on products categorization that means classifying products and providing an index in order to make them easier and fast to retrieve. Product ontology plays an important role in the construction of precise and enriched e-catalogs [9]. Ontology-based products categorization requires handling classes, properties and relationships [10] from products ontology.

Products categories tree is directly constructed from the ontology. The categorization process runs the ontology tree from products names which are individuals that together form the instances of classes (fig.2). Categories tree is created recursively by classes' generalization.

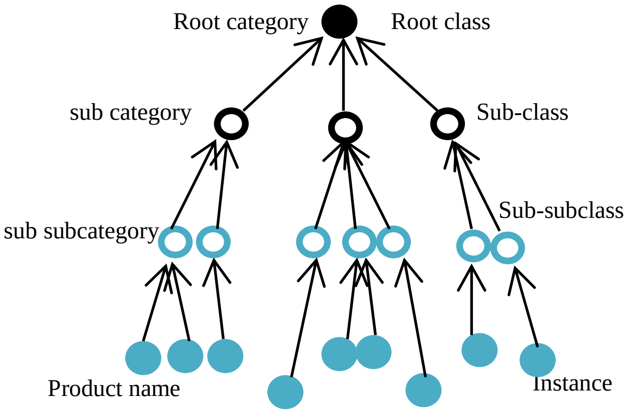

Fig 2: Categories creation from ontology

5. ARCHITECTURE DESIGN

5.1 Agent and Multi-Agent System

Agent Oriented Software Engineering has gained a lot of attention in recent years not only in academic research laboratories but also in an increasingly wide variety of software industry domains [11]. S. Sadik et al. dev a mas for Modeling high assurance agent-based Earthquake Management System [12], Yanqing Ji and al. propose a proactive MAS to deal with postmarketing drug safety surveillance. [13]. For a survey on recent research in the MAS field see [14].

5.2 General Architecture

iShopBot employs a multi-layered architecture. The diagram above (fig.3) is the general architecture used to develop the iShopBot system.

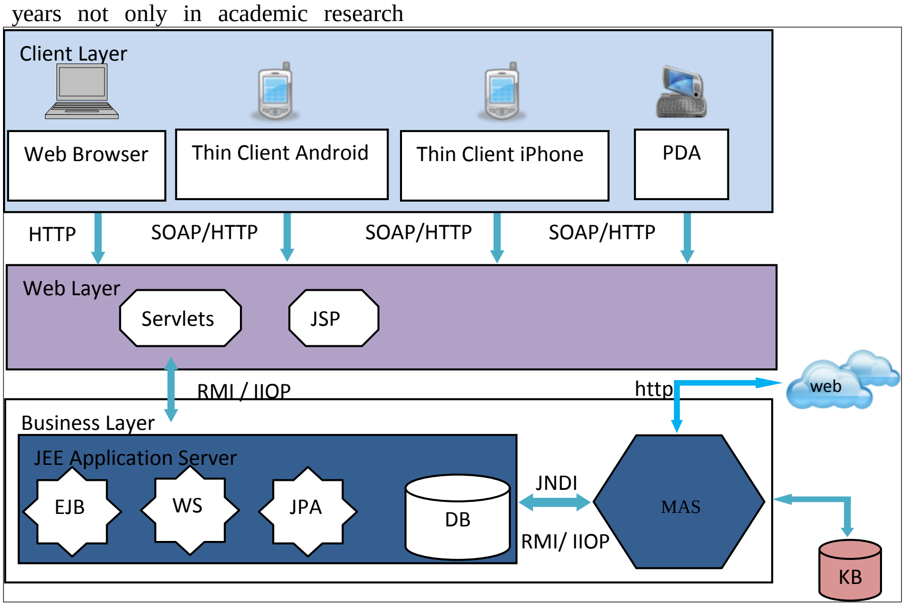

Fig 3: iShopBot General Architecture

Here is a brief description of the various components of the architecture:

• Business Layer – This layer handles requests from the web Layer. This is the layer that contains the business rules on how the iShopBot should function. It is composed of two main components:

1- JEE business application running on Glassfish application server: The JEE business application is used to deal with business logic, ensure data persistence and expose the existing web services. This application coordinates with the iShopBot MAS to carry out tasks that need intelligent skills like autonomy, reactivity communication, negotiation and natural language processing (NLP).

2- MAS running on Jade multi agents platform [15,16]: the iShopBot MAS is formed of four agent classes: Crawler Agent, Indexer Agent, Change Detector Agent and Query Handler Agent described in details in the following paragraph.

• Web Layer - The web application is the presentation or design elements of the application; such as Servlets and JSP.

• Client Layer – This layer is made up of different thin clients used by customers to access the iShopBot services: web browsers, hand held devices and PDA.

5.3 Agents class diagram

In order to express the internal architecture of each agent in the iShopBot MAS, we used Agent Modeling Language (AML)[17]. AML offers an UML extension by dealing with the particular characteristics associated with MAS like behavior abstraction, communicative interactions [13], collaboration [18], services and ontology [19] to name few.

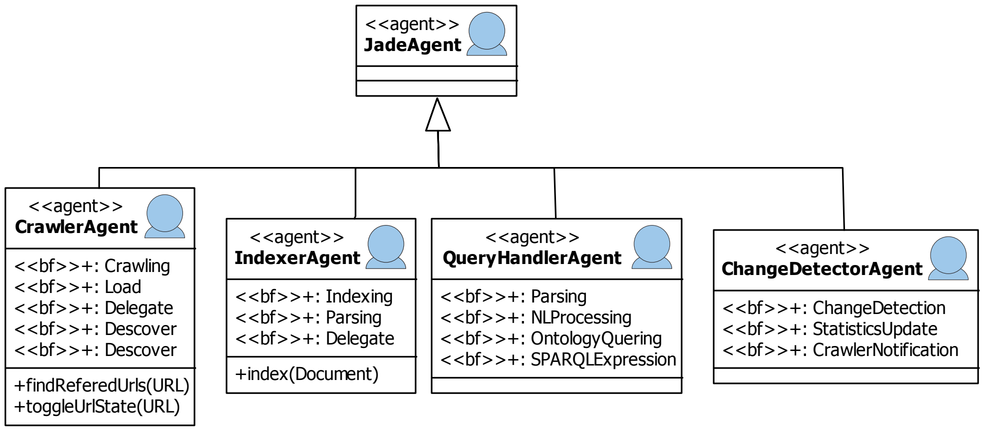

Fig 4: iShopBot Agent Diagram

5.3.1 *Crawler Agents*

In iShopBot, an intelligent agent is responsible of product data collection from e-vendors websites. E-vendors' webpages are crawled by the crawler agent (fig.5)(fig.6) starting from seeds. These seeds are e-vendors' home webpages that are primarily collected from other widespread shopping bot websites.

Once E-vendors' webpages are crawled, the agent seeks within these pages for relevant information about products such as products names, prices, delivery fees and time, and warranty fees and durations. Products relevant information is extracted from crawled webpages titles, URLs and some well-selected tags. The aim of the collected information is to enable a multi-criteria comparison.

Extracted products descriptions and characteristics are stored in a database that contains a more detailed set of information. iShopBot uses this information for automatic products' ontology learning and population.

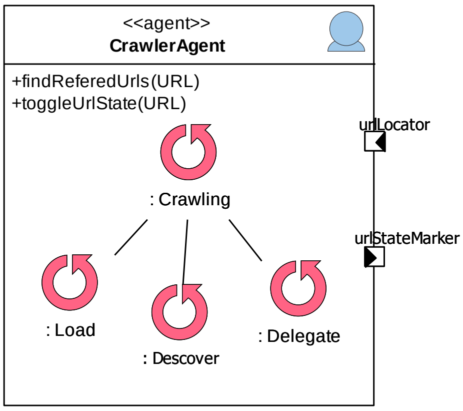

Fig 5: Crawler Agent class diagram

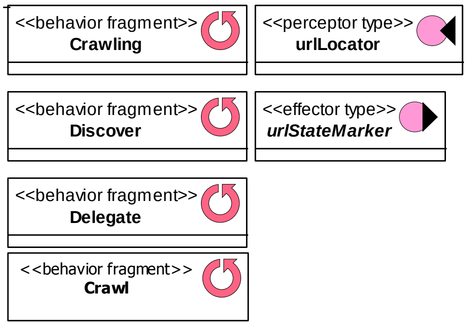

Fig 6: Crawler Agent behaviors, perceptors and affectors

5.3.2 *Indexer Agents*

This agent is responsible for indexing documents crawled by the crawler agent, it has one preceptor, two effectors and it can perform tree behaviors (fig.7), namely parsing, indexing and delegate which enable the agent to negotiate with the other indexer agents to fulfill the remaining tasks in its execution queue.

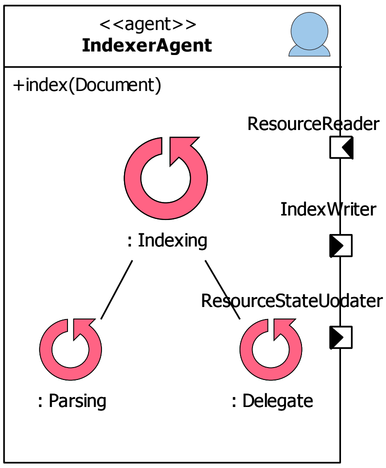

Fig 7: Indexer Agent class diagram

5.3.3 Change Detector Agent

This agent plays the role of webpages change tracker (fig.8). It updates products existing data whenever it is changed in e-vendors websites. The old values of product stored data aren't erased by the new values. The new values are used in the comparison while old ones are presented in variation graph. If a product can't be accessed several times when the agent revisits the e-vendors website, it concludes that the product is removed. In order to tune the iShopBot performances, the change detector agent records change frequency and use this statistical information to schedule next check for updates.

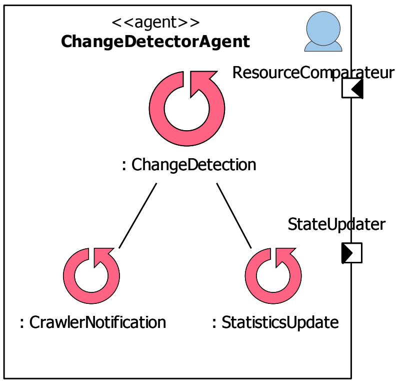

Fig 8: Change Detector Agent class diagram

5.3.4 Query Handler Agent

This agent is responsible for answering customer's queries (fig.9)(fig.10). It is equipped by various behaviors:

- Natural language processing behavior: it enables answering questions [20] expressed in natural language; the present system supports only questions in English.

- SPARQL expression: this behavior enables query handler agent to translate user queries into unambiguous ones.

- Ontology Querying: the resulted SPARQL [21] queries produced by the "SPARQL Experssion" behavior are used to retrieve and manipulate data stored in Resource Description Framework (RDF) format [22].

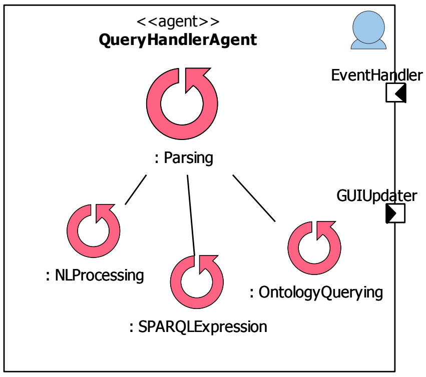

Fig 9: Query Handler Agent class diagram

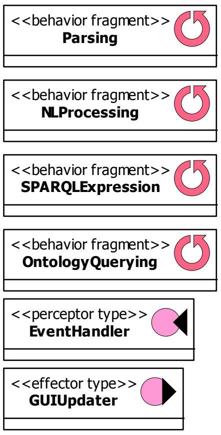

Fig 10: Query Handler Agent behaviors, perceptors and affectors

5.4 Agent Communication

5.4.1 Agent Communication

Because an agent represent an autonomous computing entity, reactive, proactive and socially habile [23], the MAS functionalities can be regarded as results of communicative acts among agents in the MAS [24]. iShopBot use ACL as a communication language among agents [25,26]. The content of each ACL message is expressed in SL content language [27]. The ACL message content is formed of a set of symbols defined in the iShopBot specific Ontology. AML offers many diagrams to represent the interactions among agents. The following diagram depicts the interaction between a crawler agent and an indexer agent (fig.11).

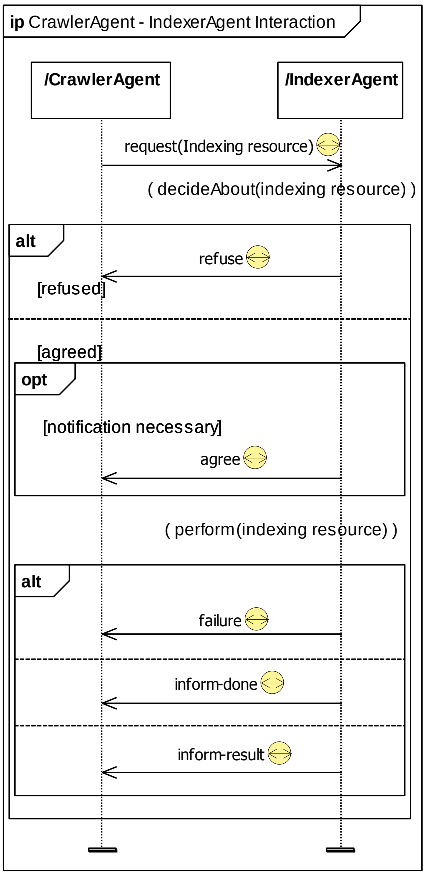
ip CrawlerAgent - IndexerAgent Interaction
/CrawlerAgent
/IndexerAgent
request(Indexing resource)
(decideAbout(indexing resource))
alt
refuse
[refused]
[agreed]
opt
[notification necessary]
agree
(perform(indexing resource))
alt
failure
inform-done
inform-result

Fig 11: interaction diagram crawler indexer

6. EVALUATION

Evaluating a shopping bot isn't a trivial task because the performance in this area isn't a measurable entity. To have an idea about iShopBot performance, we have focused on the Query Handler agent results. For this purpose, we have used two queries and compared the results in two well-known shopping bots that we will call shopbot1 and shopbot2.

The first Query was about "digital versatile disc" and the result in shopbot1 was something different than a DVD (Versatile Molle Backpack with Water Backpack Space).

iShopBot and shopbot2 have passed this test by giving results related to DVD.

The second Query was expressed as human question: "How to store my personal data on DVD?". The result in shopbot1 is (*Sorry, there are no results for how to store my personal data on dvd. A broader search for "how to store my" gives the following results*) and the first result is about "*hair bows*". Shopbot2 first result is about a "*personal cloud storage service*" while iShopBot first result is about a "*DVD burner".*

7. CONCLUSION

This paper introduced iShopBot, a shopping bot that can be called intelligent because, in one hand its Query Handler agent is based on semantic web technology. Therefore, it uses not only keywords but also their semantically related terms. It also equipped by a question answering system that answers users question expressed in the natural human language. In the other hand it is heavily based on a multi-agent system that updates data in an optimal way by calculating e-vendors data update time. iShopBot is also a deeply automated shopping bot. In fact, products data collection, products ontology construction, products categorization and products data change tracking are all automated.

Future iShopBot versions will include several updates by related to adaptations of results to user's profile adaptation taking into account user's behavior. iShopBot will provided with a support for multiplatform communication.

8. ACKNOWLEDGMENTS

Our thanks to the IbnZoh University and Cadi Ayyad University for supporting this work in part.

9. REFERENCES

[1] T. Berners-Lee, J.A. Hendler, O. Lassila 2001, The semantic web, Scientific American Magazine 284, Pages 34–43.

[2] P. Maes, R.H. Guttman, A.G. Moukas, 1999. Agents that buy and sell, Communications of the ACM 42, Pages 81–91.

[3] S. Huang, Y. Tsai 2011. Designing a cross-language comparison-shopping agent, Decision Support Systems 50, Pages 428–438

[4] G. Swamynathan, C. Wilson, B. Boe, K. Almeroth, B. Y. Zhao, 2008. Do Social Networks Improve e-Commerce? A Study on Social Marketplaces, WOSN'08, Seattle, Washington, USA,

[5] S. Staab, E. Erdmann, A. Maedche, 2001. Engineering ontologies using semantic patterns. In Proceedings of the IJCAI'01 Workshop on E-Business and Intelligent Web

[6] S. Hubner, H.Neumann, H.Stuckenschmidt, G.Schuster, T.Vogele, U.Visser, H.Wache, 2001. Ontology-Based Integration of Information-A survey of Existing Approaches

[7] P. Cimiano, J. Volker, 2005. Text2Onto - A Framework for Ontology Learning and Datadriven Change Discovery.

[8] C. Fellbaum. WordNet, an electronic lexical database. MIT Press, 1998.

[9] F. Qian, 2010. Constructing Ontology-Based Product Catalogs System for E-business, Intelligent Computation Technology and Automation (ICICTA), Pages 1027-1030

[10] D. Fensel, Y. Ding, B. Omelayenko, E. Schulten, G. Botquin, M. Brown, A. Flett, 2001. Product Data Integration in B2B E-Commerce. IEEE Intelligent Systems 16, Pages 54-59

[11] Tumer, K. and Agogino, K. 2007. "Distributed Agent-Based Air Traffic Flow Management", Distributed Agent-Based Air Traffic Flow Management. In: AAMAS: Proceedings of the 6th international joint conference on autonomous agents and multiagent system. New York, NY, USA:ACM, 1-8. isbn:978-81-904262. DOI: http://doi.acm.org/10.1145/1329434.

[12] Sadik, S. et al. 2010. Modeling high assurance agent-based Earthquake Management System using formal techniques. The Journal of Supercomputing (TJS) 52(2):97-118

[13] Ji,Y. et al. A distributed, 2010. collaborative intelligent agent system approach for proactive postmarketing drug safety surveillance. IEEE Transactions on Information Technology in Biomedicine (TITB) 14(3):826-837

[14] Onoda, T. Bekki, D. and Mc Cready, E. 2012. New Frontiers in Artificial Intelligence, Lecture Notes in Artificial Intelligence (LNAI 6797), Springer-Verlag.

[15] Su, C.J. and Wu, C.Y. 2011. JADE implemented mobile multi-agent based, distributed information platform for pervasive health care monitoring, Applied Soft Computing Journal 11 , 315–325.

[16] Bellifemine, F. et al. 2008. JADE: A software framework for developing multi-agent applications, Lessons learned. Information and Software Technology 50 ,10–21

[17] Cervenka, R. and Trencansky,I. 2007. The Agent Modeling Language - AML: A Comprehensive Approach to Modeling Multi-Agent Systems, Whitestein Series in Software Agent Technologies and Autonomic Computing

[18] Fan, X. et al. 2010. Human-Agent Collaboration for Time-Stressed Multicontext Decision Making. IEEE Transactions on Systems, Man, and Cybernetics, Part A (TSMC) 40(2):306-320

[19] Azevedo, R. et al, 2011. An autonomic ontology-based multiagent system for intrusion detection in computing environments, International Journal for Infonomics (IJI) 3 , 182–189.

[20] V. Lopez, M. Pasin, E. Motta, 2005. Aqualog: An ontology-portable question answering system for the semantic web. In: ESWC. Pages 546–562

[21] E. Prud'Hommeaux, A. Seaborne, 2008. SPARQL query language for RDF. W3C Recommnedation

[22] G. Klyne, J. J. Carroll, 2004. Resource Description Framework (RDF): Concepts and Abstract Syntax, W3C Recommendation

[23] Wooldridge, M. 2002. An Introduction to MultiAgent Systems, John Wiley & Sons

[24] Goldingay, H. Mourik, J. 2011. The effect of load on agent-based algorithms for distributed task allocation, Informtion Sciences. doi:10.1016/j.ins.2011.06.011

[25] Bagherzadeh,J. and Arun-Kumar,S. 2006. Flexible Communication of Agents based on FIPA-ACL Original Research Article, Electronic Notes in Theoretical Computer Science, Volume 159, Pages 23-39

[26] FIPA, 2001. ACL Message Structure Specification, http://www.fipa.org/specs/fipa00061/XC00061E.html

[27] FIPA, 2002. SL Content Language Specification, http://www.fipa.org/specs/fipa00008/SC00008I.html