15th Triennial World Congress, Barcelona, Spain

KNOWLEDGE SOURCE NETWORK CONFIGURATION IN E-BUSINESS ENVIRONMENT

Alexander Smirnov, Mikhail Pashkin, Nikolai Chilov, Tatiana Levashova

St.Petersburg Institute for Informatics and Automation
of the Russian Academy of Sciences
39, 14th Line, St.Petersburg, 199178, Russia
Tel.:+7(812) 328-8071, fax: +7(812) 328-0685
E-mail: smir@mail.iias.spb.su

Abstract: Today e-business requires an intensive knowledge exchange between participants of the global business information environment (e-business environment). Along with a large number of distributed knowledge sources representing knowledge in various formats this leads to appearance of a new direction in knowledge management called knowledge logistics (KL). The paper describes Knowledge Source Network configuration approach (KSNet-approach) to KL through knowledge fusion (KF) and an agent-based architecture of the system "KSNet" based on this approach. Some major subtasks of the approach are described in detail, with related examples being included. *Copyright © 2002 IFAC*

Keywords: Business process engineering, configuration management; information integration, intelligent knowledge-based systems, knowledge representation.

1. INTRODUCTION

E-Business requires intensive cooperation and open real-time standard-based information/knowledge exchange between participants of the global business information environment (e-business environment). As a result, there arises a need for acquisition of the right knowledge from distributed sources, its integration and transfer to the right person within the right context, at the right time, for the right business purpose. In this paper the aggregate of the interrelated activities listed above is referred to as Knowledge Logistics (KL). The KL is based on individual user requirements, available knowledge sources, and current situation analysis in the e-business environment. Hence, systems operating in this area must react dynamically to unforeseen changes and unexpected user needs, keep up-to-date resource value assessment data, support rapid conducting of complex operations, and deliver results to the users/knowledge customers in a personalized way.

Here described approach addresses KL through knowledge fusion (KF). KF implies synergistic use of knowledge from different sources in order to complement insufficient knowledge and obtain new knowledge (Smirnov, 2001b).

The main principles considered during development of the presented approach and a KF system based on this approach are originated from characteristics of modern e-business applications. These applications widely use ontologies as a common language for business process / enterprise modeling (Goossenaerts and Pelletier, 2001; O'Leary, 2000; Smirnov, 2001a). Thus, the described approach is focused on utilizing reusable knowledge through ontological descriptions (Guarino, 1998), with object-oriented constraint network paradigm being considered as a common knowledge notation (Smirnov, et al., 2001b). This way of knowledge representation correlates with semantic metadata representation concept of the Semantic Web project (Semantic Web, 2001).

Application of intelligent agents representing their knowledge via ontologies (Weiss, 2000) was motivated by such requirements to KF systems as flexibility, scalability, and customizability. Multi-agent system architecture, based on FIPA Reference Model (FIPA, 1997 – 2001) was chosen as a technological basis for definition of agents' properties and functions. The multi-agent system architecture is described in (Smirnov, et al., 2001a).

2. KSNET-APPROACH

Here presented approach is based on the assumptions that (i) knowledge as a complex resource is characterized by structure, cost, location, access time and life-time, and (ii) a knowledge worker is an owner of knowledge and a member of a team/group.

The KL problem in the approach is considered as a configuration of a network which includes end-users/customers, loosely coupled knowledge sources/resources, and set of tools and methods for information processing. Network of loosely coupled sources located in the information environment will be referred to as knowledge source network or "KSNet" (Fig. 1). The upper level represents a customer-oriented knowledge model based on a fusion of knowledge acquired from KSNet units/knowledge sources (KSs) constituting the lower level and containing their own knowledge models, some of which can be alternative.

3. KNOWLEDGE FUSION SYSTEM "KSNET"

Based on the KSNet-approach operational principles, the architecture and research prototype of a KF system called "KSNet" have been developed.

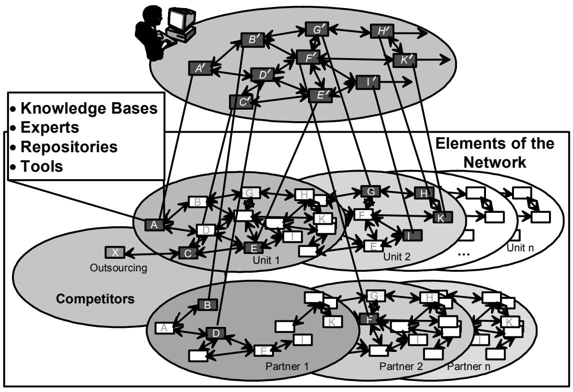

Fig. 1. Knowledge source network
(adapted from Golm and Smirnov, 2000)

The system's architecture (Fig. 2) takes into account such modern requirements to e-business applications as (i) flexibility, (ii) learning from the user, (iii) integrity, (iv) velocity, (v) open connectivity, (vi) reasoning and (vii) customizability.

The system works in terms of a common application ontology (AO) describing a problem domain and stored in ontologies library. The AO is based on domain and tasks & methods ontologies, which are also stored in the ontologies library. Each user/user group works in terms of associated expandable user request ontology and thereby with a part of AO interesting for the user/ user group. User profiles are used during interactions with users for an efficient personalized service. Every user request consists of two parts: (i) structural constituent (containing the request's terms and relations between them), and (ii) parametric constituent (containing additional user-defined constraints). For the request processing an auxiliary KSNet configuration is built. This configuration defines when and what KSs are to be used for the request processing in the most efficient way. A knowledge map including information about locations of KSs is used for this purpose. Translation between the system's and KS' notations & terms is performed using KS ontologies.

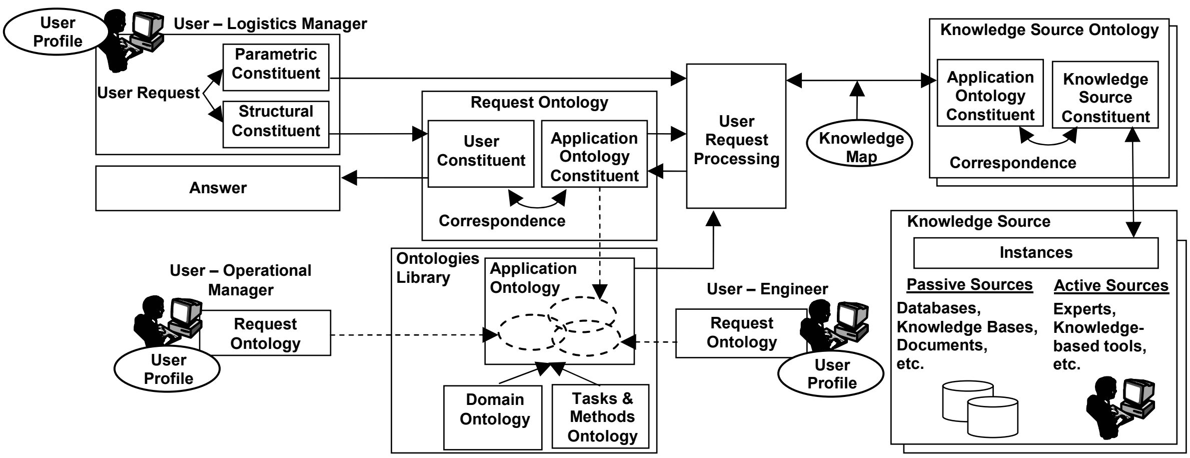

Fig. 2. Conceptual scheme of the user-oriented ontology-driven KF methodology

The main technological issues related to the system's functioning are shown in (Fig. 3). The proposed techniques addressing these issues and including (i) application ontology (AO) creation (with an automotive supply chain management used as a case study), (ii) task of efficient KSs choice for the auxiliary KSNet configuration and an application of genetic algorithm (GA) to solve it, and (iii) constraint-based configuration technique (with an automotive supply chain used as a case study) are presented below. These techniques have been tested via developed research software prototypes of the corresponding problem oriented agents.

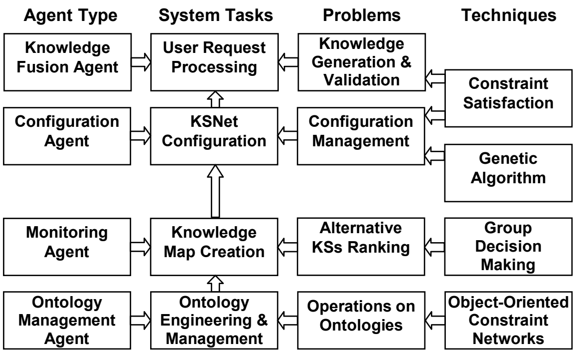

Fig. 3. Main system tasks and techniques

4. APPLICATION ONTOLOGY CREATION

AO A is based on ontologies $A_1, \ldots, A_n$ stored in the ontologies library. Designed scheme of this process includes the following sequence of ontology operations:

- *Creation*: creation of an empty structure for the application domain description;
- *Finding*: definition of the domain main terms (keywords) with subsequent search for them inside ontologies $A_1, \ldots, A_n$ of the ontologies library;
- *Slicing* (consists of the *Selection* and *Extraction* operations): selection of the keywords contexts or slices $\widehat{A}_1, \ldots, \widehat{A}_n$ in the ontologies;
- *Merging*: building a single set A' from the resulting sets $\widehat{A}_1, \ldots, \widehat{A}_n$ of the *Slicing* operation.
- *Pruning*: elimination of the set A' redundancy (the result is the set A'');
- *Modifying* (consists of the *Add*, *Delete* and *Modify* operations): making some changes in the set A'' (the result is the set A''');
- *Validation*: check for the set A''' consistency (the result is AO A).

For instance, the KF system's ontologies library contains two domain ontologies ("Management", and "Supply Chain"), and tasks & methods ontology (Fig. 4). It is necessary to create "Supply Chain Management (SCM)" ontology. In the figures presented below the hierarchical relationships ("part of") are shown as solid lines and associative relationships ("uses") are shown as dashed lines. Arrows denote references to tasks/methods (shown only for "Planning" task in Fig. 4 and omitted in other figures). In the given example the SCM keywords (shaded boxes) are: *supply chain*, *product*, *unit*, *process, resource, cost centre.* The figures below contain only classes and relationships between them, while the actual operations are performed on the entire set of ontology elements including classes, attributes, constraints/relationships, and domains.

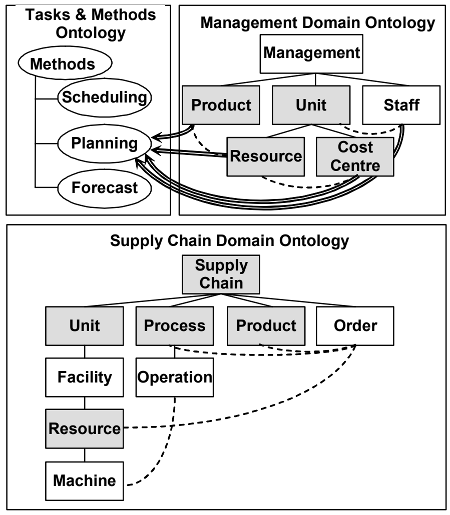

Fig. 4. Ontology library containing task & methods and domain ontologies

During the *Slicing* operation the sets of ontology elements are selected and extracted. The slice of the "Management" ontology is presented in (Fig. 5); the slice of the "Supply Chain" ontology contains all its elements.

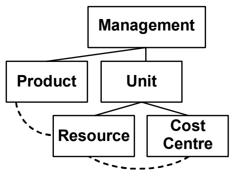

Fig. 5. Slice of the "Management" ontology

At the *Merging* operation the slices above are combined into one set. At the *Pruning* operation class "Management" is deleted because of its redundancy: the hierarchy cannot have two roots (Fig. 6).

During the *Modifying* operation experts make the following changes (the result is presented in Fig. 7): (i) deleting hierarchical relationship connecting "Cost Centre" class with "Unit" class, and adding hierarchical relationship connecting "Cost Centre" class with "Facility" class as required for more precise costs estimation in the current problem domain, and (ii) deleting classes "Operation" and "Machine" since these classes are beyond the current problem scope (relationships connecting these classes with other classes are deleted automatically).

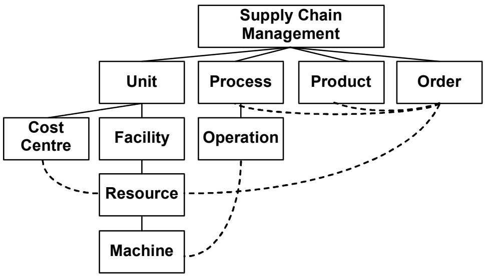

Fig. 6. Resulting set of the *Merging* and *Pruning* operations

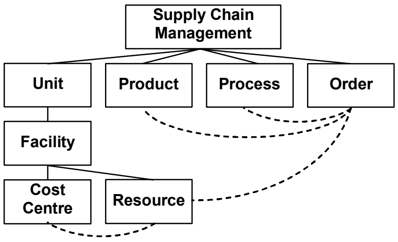

Fig. 7. Supply Chain Management ontology

Since the *Validation* operation (performed automatically and based on predefined conditions and rules) does not show any inconsistencies in the resulting set, it can be referred to as the "Supply Chain Management" AO.

5. EFFICIENT KNOWLEDGE SOURCES CHOICE

The goal of this task is a selection of KSs to be used for the user request processing in a most efficient way according to predefined criteria such as costs and/or time. The following mathematical model is a formalization of this task.

Initially the KF system includes:

AO A containing some ontology elements (OE – $\{a_j\}$), i.e. classes O, attributes Q, domains D, and constraints C of application domain.

$$A = (O, Q, D, C) = \{a_j\},\ j = 1, \dots, n \quad (1)$$

where n is the number of OEs.

KSs S_i containing some OEs $\{s_{jit}\}$ at a time instant t. Besides OEs, KS contains instances (information content I), i.e. it constitutes a constraint network $CNet(S_i)$:

$$\begin{aligned} &CNet(S_{it}) = \\ &= (O(S_{it}), Q(S_{it}), D(S_{it}), C(S_{it}), I(S_{it})) = \\ &= \{s_{jit}\},\ i = 1, \dots, m,\ t = 1, \dots, T \end{aligned} \quad (2)$$

where m is the number of KSs in the system, and T is the length of the KF system's life cycle.

Knowledge map, associating OEs of KSs with those of AO at a time instant t. Such association is denoted by a symbol "$\rightarrow$", and a statement "OE a_j is associated with KS S_{it}" is denoted by $(a_j \rightarrow S_{it})$:

$$KM_t = \{(a_j \rightarrow S_{it})\},\ a_j \in A \quad (3)$$

It is considered that for each KS its parameters such as costs, availability, access time, on-line schedule, etc. are known. KS ontology will be defined as an association of KS' elements with AO's elements:

$$A(S_{it}) = \{(a_j \rightarrow s_{jit})\} \quad (4)$$

When a user request R is received by the system it is decomposed into a set of subrequests r_k, which then are associated with the AO's OEs (i.e. translated into the system's terms). This association is contained in the request ontology $A(R)$. When these operations are completed the request translated and decomposed into subrequests associated with the AO's OEs will be obtained (denoted by R'):

$$R = \{r_k\} \quad (5)$$

$$A(R) = \{(r_k \rightarrow a_j)\},\ r_k \in R,\ a_j \in A \quad (6)$$

$$R' = \{a_j\},\ \forall a_j\ \exists (r_k \rightarrow a_j) \in A(R) \quad (7)$$

When the operations above are completed a set of feasible decisions of the task Dec_R can be written as:

$$Dec_R = \{dec_R\},\ dec_R = \{(r_k \rightarrow s_{jit})\} \quad (8)$$

Costs *Cost* or time *Time* required for request processing can be used as a criteria of the decision's effectiveness:

$$Cost = f_{Cost}(dec_R) = \sum_{s_{jik} \in dec_R} f_{Cost}(s_{jik}) \quad (9)$$

$$Time = f_{Time}(dec_R) \quad (10)$$

Also, an overall index of effectiveness *Eff* including estimations of both costs and time can be considered (multicriteria optimization). For instance, normalized values of cost and time (superscript N) functions can be summarized using weights w_{Cost} and w_{Time}:

$$\begin{aligned} &Eff = f_{Eff}(dec_R) = \\ &= f'_{Eff}(f_{Cost}(dec_R), f_{Time}(dec_R)) = \\ &= w_{Cost} \cdot f^N_{Cost}(dec_R) + w_{Time} \cdot f^N_{Time}(dec_R), \\ &w_{Cost} + w_{Time} = 1 \end{aligned} \quad (11)$$

Decision is considered effective (denoted by dec_R^{eff}) if the value of goal function, e.g., (11), is minimal with the constraints (1 - 8) being true:

$$\begin{gathered} dec_R^{eff} \in Dec_R, \\ \forall dec_R \in Dec_R,\ f_{Eff}\left(dec_R^{eff}\right) \leq f_{Eff}\left(dec_R\right) \end{gathered} \tag{12}$$

To solve this task an application of GA is proposed. A feasible static decision dec_R represents a chromosome and has the following structure:

$$dec_R = \left\{dec_{k,i}^R\right\} \tag{15}$$

where each $dec^R_{k,\,i}$ is a Boolean variable equal to 1 if KS_i is used for obtaining OE_k or to 0 otherwise. Hence, dec_R represents a binary matrix (Fig. 8), whose rows are considered as genes for GA.

$$\begin{array}{c|cccccc}
 & KS_1 & \ldots & KS_i & \ldots & KS_m \\
\hline
OE_1 & dec_{1,1}^R & \ldots & dec_{1,i}^R & \ldots & dec_{1,m}^R \\
\ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
OE_k & dec_{k,1}^R & \ldots & dec_{k,i}^R & \ldots & dec_{k,m}^R \\
\ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
OE_n & dec_{n,1}^R & \ldots & dec_{n,i}^R & \ldots & dec_{n,m}^R
\end{array}$$

Fig. 8. Structure of a feasible decision used in GA

To investigate an efficiency of GA a set of experiments with a basic GA for tasks of different dimensions have been performed, with KSs' parameters and knowledge maps being randomly generated. The results indicate that the number of required calculations for obtaining a quasi-efficient decision using even basic non-optimized GA is smaller than that in the method of exhaustive search. Fig. 9 represents the ratio of calculations number for the method of exhaustive search to that for the GA. As the task dimension grows this improvement grows nonlinearly.

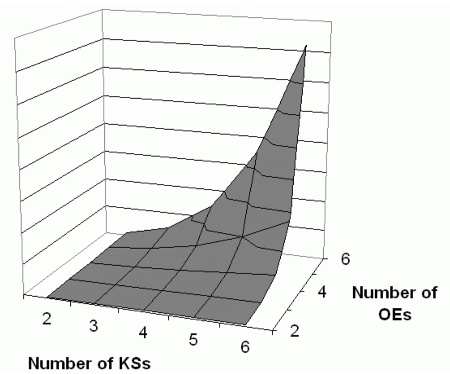

Fig. 9. Efficiency improvement due to GA application

6. PRODUCTION CONFIGURATION USING CONSTRAINT SATISFACTION TECHNIQUE

This section illustrates a constraint-based configuration task via a configuration example for a hypothetic vehicle with customer-defined features and cost constraints (Smirnov, 1999), and a supply chain for its production.

In the first part of the example the vehicle consists of three main components: (i) body, (ii) engine, and (iii) transmission (Table 1). It is also known that not all options presented can be combined: for instance, engine 2.0L cannot be installed on Sedan. The incompatible options are presented in Table 2.

Table 1. Cost data for vehicle components

Component class	Component name	Component price, $
Body	Sedan	10,990
	Hatchback	11,320
	Wagon	12,300
Engine	1.8L	2,300
	2.0L	2,500
Transmission	5-sp. automatic	820
	4 sp. manual	550

Table 2. Incompatible vehicle configuration options

Option A	Option B
Sedan body	2.0L engine
Wagon body	1.8L engine

To implement the constraint network based on the application ontology the features of ILOG Configurator are used (ILOG, 2001). Since ILOG Configurator represents the task to be solved in the object-oriented form, the built AO can be represented via ILOG means without significant modifications.

In the second part of the example considering a supply chain the production process consists of three parallel tasks: (i) body production, (ii) engine production, and (iii) transmission production. Facilities are the plants, with known capacities and such characteristics as production cost and time. The goal is cost minimization within time limit or time minimization within cost limit (Fig. 10).

During the task solving process the system builds a constraint network based on the related information from the system's repository and the user's request; acquires required data from appropriate sources (emulated via several databases); searches for feasible and efficient decision; and delivers the results to the user.

7. CONCLUSIONS

In the face of globalization in e-business and worldwide increasing competition the knowledge logistics can by very powerful to enable collaboration between global business environment members.

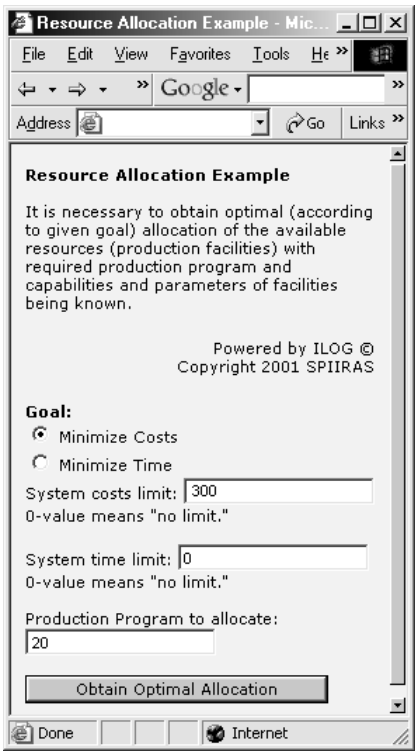

Fig. 10. Screen form example for supply chain configuration prototype

Some of the main advantages of the developed KSNet-approach in a scope of here described fragments are: (i) using ontologies libraries with top-level ontology facilitates problem domain description, (ii) KSNet configuration using genetic algorithm provides for a faster knowledge search and an ability to take into account additional user constraints for request processing such as time, costs, etc., and (iii) constraint networks can be used as models for a wide class of e-Business application problems description (e.g. product configuration based on the customer's requirements).

ACKNOWLEDGMENTS

The paper is due to the research carried out as a part of the Partner Project # 1993P funded by EOARD, and as a part of the project # 4.4 of the research program # 18 "Intelligent Computer Systems" of the Russian Academy of Sciences. Some examples were developed using software granted by ILOG Inc.

REFERENCES

Foundation for Intelligent Physical Agents (FIPA) Documentation (2001), http://www.fipa.org.

Golm F. and A.Smirnov (2000). ProCon: Decision Support for Resource Management in a Global Production Network. In: *Proceedings of the 13 th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems (IEA/AIE)*. New Orleans, Louisiana, USA, pp. 345—350. Lecture Notes in Computer Science, **1821**.

Goossenaerts J. and C.Pelletier (2001). Enterprise Ontologies and Knowledge Management. In: *Proceedings of the 7th International Conference on Concurrent Enterprising "Engineering the Knowledge Economy through Co-operation"*, pp. 281—285. Bremen, Germany.

Guarino N. (1998). Formal Ontology and Information Systems. In: *Proceedings of FOIS'98*, pp. 3—15. Trento, Italy. IOS Press, Amsterdam, Holland.

ILOG Configurator (2001). http://www.ilog.com/-products/configurator.

O'Leary D.E. (2000). Different Firms, Different Ontologies, and No One Best Ontology. *IEEE Intelligent Systems*, **September/October**, pp. 72—78.

Semantic Web (2001). http://www.semanticweb.org.

Smirnov A. (1999). Virtual Enterprise Configuration Management. In: *Preprints of the 14th World Congress of IFAC International Federation of Automatic Control*, **A**, pp. 337—342. Beijing, China.

Smirnov A. (2001a). Ontology Drive Virtual Production Network Configuration: a Concept and Constraint-Object-Oriented Knowledge Management. In: *Proceedings of the Third International Conference on Enterprise Information Systems (ICEIS 2001)*, pp. 345—352. Setubal, Portugal.

Smirnov A. (2001b) Profile-Based Configuring of Knowledge Supply Networks in the Global Business Information Environment. *Proceedings of the 2001 IEEE International Conference on Systems, Man and Cybernetics "e-Systems and e-Man for Cybernetics in Cyberspace"*, pp. 977—982, Tucson, Arizona.

Smirnov A., Pashkin M., Chilov N., and T. Levashova (2001). Multi Agent Architecture for Knowledge Fusion from Distributed Sources. In: *Proceedings of 2nd International Workshop of Central and Eastern Europe on Multi-Agent Sytems (CEEMAS 2001).* Krakow, Poland, pp. 403—412.

Smirnov A., Pashkin M., Chilov N., and T. Levashova (2001). Ontology Management in Multi-Agent System for Knowledge Logistics. *Proceedings of the International Conferences on Info-tech & Info-net (ICII2001-Beijing)*, Beijing, CHINA. (Electronic Proceedings).

Weiss G. (2000). *Multiagent Systems: a Modern Approach to Distributed Artificial Intelligence* (ed. by G.Weiss). The MIT Press, Cambridge, Massachusetts, London, **8**.