Towards Methodology for Design of Context-Aware Decision Support Systems based on Knowledge Fusion Patterns

Alexander Smirnov and Tatiana Levashova

St. Petersburg Institute for Informatics and Automation
of the Russian Academy of Sciences,
39, 14th line, St. Petersburg, 199178, Russian Federation
{smir, tatiana.levashova}iias.spb.su

Abstract. A pattern-based methodology for design of context-aware decision support systems is proposed. The methodology is based on knowledge fusion patterns for the knowledge fusion processes occurring at different stages of a context-aware decision support system. The methodology as it stands focuses on the context-aware stages of the system. For the knowledge fusion processes ongoing at these stages the patterns' elements relevant to the methodology are specified. Usage of the patterns to propose system functionality depending on the user needs is demonstrated.

Keywords: context aware decision support, ontology-based context, knowledge fusion, knowledge fusion patterns, pattern-based methodology.

1 Introduction

Various methodologies for information system development [1], [2], [3], [4], [5] consider information requirements as an important element of the system design. The methodologies aim is to propose the system functionality suitable to the user.

Originally, the research presented in this paper was devoted to revealing of knowledge fusion patterns for the processes occurring in a context-aware decision support system (DSS) [6], [7]. The importance of knowledge fusion in context-aware DSSs is reasoned by the intention of this technology. The objective of knowledge fusion is to integrate information and knowledge from multiple sources into some common knowledge that may be used for decision making and problem solving or may provide a better insight and understanding of the situation under consideration [8], [9], [10], i.e. knowledge fusion facilitates situational awareness.

The revealed patterns describe the knowledge fusion effects discovered in the DSS, specify states of the knowledge sources involved in knowledge fusion, and lift the effects at the ontology level. The main contribution of the present work is a pattern-based specification of the DSS' requirements to the knowledge sources involved in the processes ongoing in this DSS. The specification is intended to describe the system functionality.

Although the patterns can be used to describe the DSS' requirements ensuring the overall functionality of the system, this paper covers only context-aware phase of the DSS. It is motivated by the ongoing research; the methodology has not been developed to the full extent. It is believed that the ideas presented in this paper provide a general conception according to that the methodology will be developed.

The rest of the paper is as follows. Section 2 gives an overview of possible knowledge fusion effects and introduces the elements of the pattern description language. Section 3 describes the conceptual framework that the DSS is based upon, the stages of the DSS scenario, and presents the knowledge fusion effects produced at these stages. The pattern-based methodology is presented in Section 4. The main results and the future research are discussed in the Conclusion.

2 Knowledge Fusion Patterns

In this Section knowledge fusion patterns are revealed for the knowledge fusion processes occurring in the context-aware DSS. Knowledge fusion effects found in this DSS indicate presence of such processes.

2.1 Knowledge Fusion Effects

The main feature of knowledge fusion lies in creation of a synergic effect from the integration of information/knowledge. Based on the analysis of knowledge fusion studies, a number of knowledge fusion processes producing different effects have been distinguished:

- Intelligent fusion of huge amounts of heterogeneous data / information from a wide range of distributed sources into a form which may be used by systems and humans as the foundation for problem solving and decision making [8], [9]. The intelligence assumes consideration of the semantic contents of the sources being fused.
- Integration of knowledge from various knowledge sources resulting in a completely different type of knowledge or new idea how to solve the problem [11], [12], or integration of different types of knowledge (domain, procedural, derived, presentation, etc.) resulting in a new knowledge type [10].
- Combining knowledge from different autonomous knowledge sources in different ways in different scenarios, which results in discovery of new relations between the knowledge from different sources or/and between the entities this knowledge represents [13], [14].
- Integration of multiple knowledge sources into a new knowledge object, which is a new knowledge source [15], [16].
- Inference of explicit knowledge from information/knowledge hidden in knowledge sources being integrated or fused [17].
- Re-configuration of knowledge sources to achieve a new configuration with new capabilities or competencies or knowledge exchange to improve capabilities or competencies through learning, interactions, discussions, and practices [18].
- Involving knowledge from various sources in problem solving, which results in a solution [6].

From the analysis above it is noticed that different processes can produce the same effect, and different effects may be outcomes of the same process. The following not mutually exclusive kinds of new knowledge produced as the knowledge fusion effects are distinguished: 1) a new type of knowledge; 2) a new knowledge source; 3) a new knowledge created from data/information; 4) a new knowledge about the conceptual scheme (new relations, concepts, properties, etc.); 5) a new explicit knowledge; 6) new capabilities/competencies of a knowledge object (an object that produces or contains knowledge); 7) a new problem solving method or idea how to solve the problem; 8) a solution for the problem.

Almost all the listed effects have been found in the DSS. An exception concerns appearance of new ideas how to solve the problem. Such ideas may come as a result of conscious interactions, discussions, and practices. These issues are not considered in the research. Below, dimensions proposed to the generalization of the knowledge fusion processes and elements of a pattern specification language are discussed.

2.2 Pattern Language

Knowledge fusion involves multiple sources in the integration processes. Autonomies and structures of such sources have been chosen as the concepts in terms of which knowledge fusion patterns are revealed. In the context-aware systems integration of information/knowledge refers to the process of integration of their conceptual structures. Therefore, source's structure is an obligatory concept taken into account by the integration. In this research, by sources' structures the conceptual structures that represent the knowledge in the knowledge sources are meant.

Autonomy creates awareness of the reliability of the information/knowledge represented in the sources. The consideration of the knowledge source autonomy concept is reduced to the detection of relations existing between knowledge sources regardless of their structures. Autonomous knowledge source is an independent source having no relationships with other sources. Such source can get changed at any time, at that, the changes in this source produce no changes in other sources. On the contrary, non-autonomous source is linked to other (non-autonomous) sources. Changes in a non-autonomous source produce appropriate changes in the related sources.

The DSS operates in a dynamic environment. Information and knowledge represented in environmental sources that are related to the internal system sources (environmental and system sources are non-autonomous) are considered to be more reliable than information/knowledge represented in the autonomous environmental sources. An argument in favor of this is any changes in the linked (non-autonomous) environmental sources are reflected in the system sources.

The patterns measure knowledge fusion outcomes in terms of preservation/change of the structures and autonomies of the initial and target knowledge sources, and in terms of the effects the knowledge fusion processes produce in the DSS.

Initial knowledge sources are the sources that are integrated leading to the emergence of a new knowledge (producing some knowledge fusion effect). The sources resulting from the knowledge fusion or enclosing the knowledge fusion result are referred to as target knowledge sources. The environmental sources (below, resources) include sources of data/information/knowledge. In this sense the environmental resources belong to the collection of knowledge sources.

The knowledge fusion patterns are described using a pattern description language [7]. The detailed presentation of the patterns elements is as follows:

Name: a name to refer to the pattern
Problem: a problem the knowledge fusion process solves
Solution: a meaningful description of the knowledge fusion process
Initial knowledge source(s): knowledge sources(s) that are integrated leading to producing some knowledge fusion effect
Target knowledge source(s): knowledge sources(s) resulting from the knowledge fusion or enclosing the knowledge fusion result
Related pattern (may be omitted): an alternative pattern that can be used instead of the described one or in parallel or after termination of the described pattern
Exception (may be omitted)**:** a description of cases when the pattern is not applicable
Autonomy pre-states: the degree of autonomy of knowledge sources before the knowledge fusion process. Three degrees are provided for: autonomous, non-autonomous, and n/a (for a non-existing knowledge source)
Effect in DSS: the effect the knowledge fusion process produces in the DSS
Effect in ontology terms: ontology-based generalization of the effect produced
Post-states: the knowledge source autonomy and structures preservation degrees after the knowledge fusion process completes. For the knowledge source autonomies the degrees introduced in pre-state descriptions are kept on. Three degrees of knowledge object structure preservations are provided for: preserved, changed, and new (for a new knowledge object)
Schematic representation: the knowledge fusion process represented schematically
DSS stage: the stage of the DSS scenario where the knowledge fusion process occurs.

In this work one of the possible pattern applications, which is the pattern-based methodology for design of context-aware DSSs, is offered. Prior to present this methodology, the conceptual framework the DSS is based upon and the context-aware stages of the DSS scenario are described.

3 Context-Aware Decision Support System

The DSS is intended for support of decisions on planning emergency response actions. A two-level representation of emergency situation is used in the DSS. At the first level the situation is represented by *abstract context* that is an ontology-based intensional model of the situation. At the second level the emergency situation is represented by *operational context* that is an instantiation of the abstract context for the actual circumstances.

Environmental resources produce the operational context and solve the problem of planning emergency response actions based on this context. For this the resources organize a resource network. Nodes of this network represent the resources; network arcs signify the order of the nodes execution.

The problem solution is a set of alternative emergency response actions feasible in the current emergency situation. The decision maker chooses an alternative from the set of feasible ones. The chosen alternative is considered as the decision. According to this decision the response actions are undertaken.

Once the interactions of the decision maker with the DSS have been finished, the abstract context, the operational context, the decision, and the resources' representations are saved in a context archive. At that, the operational context and the resources' representations are saved in their states at the instant of the alternatives generation.

The DSS scenario follows two main phases: preliminary and executive. At the preliminary phase an application ontology (AO), which describes knowledge of the emergency management domain, is built. This ontology specifies knowledge to describe the emergency situations happening in this domain along with problems requiring solutions in these situations. The AO is a knowledge source fusing two types of knowledge: domain and problem-solving. The executive phase concerns support of the decision maker with alternative decisions, decision implementation, and archiving. The executive phase is the focus of this paper since at this phase context-aware functions of the DSS come into operation. Several stages are distinguished at this phase. At each of them one or more knowledge fusion effects manifest.

3.1 Context-Aware Stages of DSS

This Section focuses on the knowledge fusion processes going on at the context-aware stages of the DSS and the knowledge fusion effects produced at these stages. The processes are generalized in the patterns terms. In the paper only pattern elements that are relevant to the discussion of the proposed methodology are presented. They are the states for the autonomies and structures of the knowledge sources.

3.1.1 Abstract Context Creation

The abstract context represents knowledge relevant for decision making in the emergency situation. The represented knowledge is captured from the AO based on the type of emergency event. A (smart) sensor reads the type of event and sends it to the DSS or the user enters it in the system.

The knowledge fusion effect produced at the stage of abstract context creation is a new knowledge source of the same type as the initial knowledge source (the AO). The AO preserves its structure and autonomy; the abstract context becomes an autonomous knowledge source with a proper structure. The processes going on at the stage of abstract context refinement are generalized by the *simple fusion* pattern (Table 1).

Table 1. Simple fusion

Pattern element	Initial knowledge source	Target knowledge source
Knowledge source	application ontology	abstract context
Autonomy pre-state	autonomous	n/a
Structure post-state	preserved	new
Autonomy post-state	autonomous	autonomous

3.1.2 Abstract Context Refinement

In the abstract context the captured knowledge may result in discovery of new relationships between the knowledge unrelated in the AO. These relationships are the result of deductive inference. Generally, any kind of knowledge representation items can be inferred. The inferred items are considered as a knowledge fusion effect that is the new knowledge. This knowledge is introduced in the abstract context, hereby changing its structure. In the case of the abstract context refinement the abstract context plays the roles of the initial and target knowledge source at the same time. The processes taking place at the stage of the abstract context refinement are generalized by the *extension* pattern (Table 2).

Table 2. Extension

Pattern element	Initial knowledge source	Target knowledge source
Knowledge source	abstract context	abstract context
Autonomy pre-state	autonomous	autonomous
Structure post-state	changed	changed
Autonomy post-state	autonomous	autonomous

3.1.3 Abstract Context Reuse

The abstract contexts are reusable components of the DSS. The reuse of an abstract context in settings when the available resources are not intended to solve the problems specified in this context may result in finding alternative resources. For instance, one unavailable method can be substituted for a sequence of methods providing by the available resources. This leads to a new configuration of the resource network.

At the stage of abstract context reuse the knowledge fusion effect is twofold: a new (alternative) problem solving method and a new configuration of the resource network. A specification of the new method(s) is introduced in the abstract context. The context structure is changed, at that this context remains autonomous. The autonomies and structures of the resources representing the new method(s) are preserved.

In the case of abstract context reuse the knowledge fusion pattern is nested. The main pattern "*configured fusion*" (Table 3) includes the *extension* pattern. The *extension* pattern corresponds to introducing the new specification in the context.

Table 3. Configured fusion (the main pattern)

Pattern element	Initial knowledge source	Target knowledge source
Knowledge source	abstract context	resource network
Autonomy pre-state	autonomous	autonomous
Structure post-state	changed	preserved
Autonomy post-state	autonomous	autonomous

3.1.4 Operational Context Producing

An operational context is produced through the semantic fusion of data/information from multiple environmental resources within the ontological structure of the abstract context. Initially the operational context is a copy of the abstract context. As soon as the resources start instantiating this copy, they lose their autonomies. When the copy is fully instantiated it becomes the operational context. At that, information from the resources is constantly coming into this context. Therefore, the operational context and the environmental resources are related over the period of decision making and implementation. In practice, the operational context represents the map of the area around the emergency event where the situation dynamic is represented (the mobile responders are moving, the traffic situation is changing, etc.).

The knowledge fusion effects had at the stage of the operational context producing are 1) the operational context is *a new knowledge source*; 2) this context is *a knowledge source created from data/information*; and 3) the operational context represents *knowledge of a new* dynamic *type*. The abstract context preserves its structure and autonomy when the operational context is produced. The operational context is a new non-autonomous knowledge source. The *instantiated fusion* pattern (Table 4) generalizes the processes going on at the stage of operational context producing.

Table 4. Instantiated fusion

Pattern element	Initial knowledge source	Target knowledge source
Knowledge source	abstract context	operational context
Autonomy pre-state	autonomous	n/a
Structure post-state	preserved	new
Autonomy post-state	autonomous	non-autonomous

3.1.5 Problem Solving

As it is said above, the result of problem solving is a set of feasible emergency response plans. An emergency response plan is a set of emergency responders with required helping services, schedules for the responders' activities, and transportation routes for the mobile responders. In the plans earlier independent entities become related, i.e. new relations between these entities have arisen.

The plans are represented in the picture of the operational context. In this way, the operational context and the results of problem solving are fused forming, at that, a new knowledge source. This new source represents knowledge of a new type (the instantiated knowledge fused with the solution set).

At the problem solving stage the operational context dissolves within the new knowledge source and does not preserve the structure and autonomy. At time of alternatives generation and decision making the environmental resources and the operational context are related (non-autonomous). As soon as the decision has been made the new knowledge source and the environmental resources become autonomous. The knowledge fusion effects produced at this stage are 1) new relations between entities, 2) a problem solution, and 3) a new knowledge source of a new type. Table 5 presents a fragment of the *flat fusion* pattern for the processes at the problem solving stage.

Table 5. Flat fusion

Pattern element	Initial knowledge source	Target knowledge source
Knowledge source	operational context	knowledge source fusing the operational context and the set of alternatives
Autonomy pre-state	non-autonomous	n/a
Structure post-state	changed	n/a
Autonomy post-state	n/a	autonomous

3.1.6 Decision Implementation

The decision is a solution that the decision maker has chosen from the set of alternative ones. This decision is made at a certain time instant. The situation may change from the moment the decision was made to the moment of its implementation. The responders whom the decision is delivered may be unable to implement it in the changed circumstances. In some cases, the activities assigned to the responders who become unable to operate can be delegated to or redistributed between other responders participating in the decision implementation. As a result of this, the responders that are ready to take the assignments gain new capabilities / competencies.

For instance, an emergency team trained to rescue operations has failed in the course of actions because of a road destruction, ambulance blockage, etc. In certain cases these operations can be delegated to available teams. In the DSS the emergency responders are represented by their profiles. In the case of consent, the plan (the decision) is adjusted accordingly and the profiles of the teams agreed to take part in the rescue operations are extended with the new capability.

At the time of the decision implementation, the responders taking part in the response plan are not autonomous. Moreover, in the course of the response actions the structures of their profiles as well as the decision structure may change. The changed decision structure results in changing the structure of the knowledge source containing the set of solutions. This knowledge source is not autonomous until the decision is implemented. The knowledge fusion effect produced at the decision implementation stage consists in gaining new capabilities / competencies by the emergency responders. The *adaptation* pattern (Table 6) generalizes this case.

Table 6. Adaptation

Pattern element	Initial knowledge sources	Target knowledge sources
Knowledge source	• knowledge source representing the decision • profiles of the emergency responders	
Autonomy pre-state	non-autonomous	non-autonomous
Structure post-state	changed	changed
Autonomy post-state	non-autonomous	non-autonomous

3.1.7 Archival Knowledge Management

The stage of archival context management deals with the management of knowledge contained in the archived components. The main intention of such management is inference of new knowledge based on the accumulated one.

For example, an emergency team participated in different emergency response actions. Some operational contexts in which this team appeared and then participated in corresponding actions represent the same hospital. Based on a comparative analysis of these operational contexts it can be judged that most probably the team is a part of the hospital found together with this team in different contexts. The *part-of* relation between the emergency team and hospital is the new revealed relation. The revealing of a new knowledge based on a set of observations is a kind of inductive inference.

In the archive, the operational contexts, the representations of environmental resources, the responders' profiles, and the knowledge object representing the decision are related. I.e., in the archive all the listed knowledge sources are non-autonomous. As a result of archival knowledge management, new knowledge about conceptual schemes of the archived knowledge sources can be inferred. This new knowledge gets specified in the AO. As a result of this, the structure of the AO gets changed but its autonomy is preserved. The knowledge fusion effect produced is a new knowledge about the conceptual scheme. Table 7 shows a fragment of the *historical fusion* pattern for the processes occurring at the stage of archival knowledge management.

Table 7. Historical fusion

Pattern element	Initial knowledge sources	Target knowledge source
Knowledge source	operational contexts	application ontology
Autonomy pre-state	non-autonomous	autonomous
Structure post-state	preserved	changed
Autonomy post-state	non-autonomous	autonomous

4 Pattern-Based Methodology

The offered methodology follows four steps: 1) specification of the system information requirements; 2) capturing the user requirements; 3) matching the system requirements against the user requirements; 4) finding available system functionality.

The proposed patterns enable to formulate the system requirements in terms of the patterns' inputs/outputs (Table 8). Further on in this paper only the patterns' inputs are taken into account to formulate the requirements. The patterns' inputs and outputs jointly are planned to be used to track the information flows across the DSS' stages and between the patterns. These flows manifest interrelationships between the stages and thereupon allow ones to specify explicitly in the methodology what output scenario components can serve as the input at what stage.

The system requirements (Table 8) are formulated for the particular DSS. The patterns' parameters are presented in the way they are used in this DSS. These parameters generalized are presented in Table 9 (the columns "System requirements"). These columns formalize possible parameters' values for general cases and formulate general conditions for the patterns applicability. In the table the following notation is used: “a” – autonomous knowledge objects, “na” – non-autonomous, “m” – modifiable, “nm” – not-modifiable, “/” – logical OR, “&” – logical AND. The resource network is considered as a single knowledge object. Modifiable resource network means that this network is reconfigurable, at that any changes inside the network nodes are not supposed (the structures of the resource organizing the network are not changed).

User requirements are demonstrated by an example. For instance, the user does not possess any AO. Though, he/she has an unalterable abstract context representing the abstract situation this user usually deals with and a set of resources authorized for his/her needs. As well, this user can manage the actors’ profiles as at he/she discretion. The example of the user requirements is presented in the column "User requirements" (Table 9). "Not defined" parameter value means that the user has no specific requirements to the input component. The system designers can manipulate such a component by their choice. Here it is supposed that the system provides the user with the components resulted in the system scenario execution with the parameters required for the overall system functionality.

The system requirements are matched against the user requirements to determine the patterns applicability for this user (the column "Pattern applicability" of Table 9). From Table 9 seen, that the fully applicable patterns are *flat fusion* and *adaptation*.

Table 8. DSS requirements

Stage	Input	Output	Functionality	Pattern
Abstract context creation	Autonomous AO	Autonomous abstract context	Creation of a non-instantiated ontology-based model of the situation	Simple fusion
Abstract context refinement	Autonomous modifiable abstract context	Autonomous abstract context	Inference of new (contextual) knowledge	Extension
Abstract context reuse	• Autonomous modifiable abstract context • Autonomous reconfigurable resource network	• Autonomous abstract context • Autonomous resource network	Reconfiguration of the resource network according to the current circumstances	Configured fusion
Operational context producing	• Autonomous abstract context • Resources able to lose their autonomies	• Non-autonomous modifiable operational context • Non-autonomous resource network	Creation of a near real-time picture of the situation	Instantiated fusion
Problem solving	• Non-autonomous modifiable operational context • Resources able to lose their autonomies	• New autonomous knowledge source • Autonomous resource network	Providing the decision maker with a set of alternative decisions	Flat fusion
Decision implementation	• Non-autonomous modifiable knowledge source representing the decision • Non-autonomous modifiable actors' profiles		Gaining new capabilities / competencies by actors	Adaptation
Archival context management	• Non-autonomous operational contexts • Autonomous modifiable AO	Autonomous AO	Inductive inference of new knowledge	Historical fusion

The *configured fusion, instantiated fusion,* and *historical fusion* patterns are applicable partly. The applicability of the *configured fusion* and *instantiated fusion* patterns covers management of the resource network. The *historical fusion* pattern allows for some inductive inference. The remaining patterns are inapplicable.

The analysis of the applicable patterns results in the following system functionality. The system can reuse the existing abstract context. As introducing any knowledge into this context is not allowed, this context can be reused if the previously used set of environmental resources is available. At that, any new network configurations are senseless as they cannot be specified in the context. Having the abstract context and the resource network the system can provide the user with an operational context (the dynamic picture of the situation) and a set of feasible decisions in this situation. As the user is able to manage the actors' profiles, the system provides he/her with the ability to manage the decision implementation. The operational contexts produced based on the reusable abstract context can be archived, inductive inference over them can be supported but the inference results cannot be retained.

Table 9. Pattern-based requirements

Pattern	System requirements		User requirements	Pattern applicability
	Input(s)	Parameter value	Parameter value	
Simple fusion	AO	a/na/m/nm	unavailable	n/a
Extension	Abstract context	a/na&m	nm	n/a
Configured fusion	Abstract context	a/na&m	nm	n/a
	Resource network	a/na/m/nm	a/na&m	applicable
Instantiated fusion	Abstract context	a/na/m/nm	nm	n/a
	Resource network	na&m/nm	a/na&m	applicable
Flat fusion	Operational context	na&m	not defined = na&m	applicable
	Resource network	na&m/nm	a/na&m	applicable
Adaptation	Decision	na&m	not defined = na&m	applicable
	Actors' profiles	na&m	na/a&m	applicable
Historical fusion	Operational contexts	a/na/m	not defined = a/na/m	applicable
	AO	a/na&m	unavailable	n/a

The proposed functionality means that the user can deal with the situation he/she usually deals with, solve repetitive problems with different values for the problems' variables, and manage the decision implementation. For instance, applying to the emergency management domain, the system can support decisions on the emergency situations caused by one and the same type of event. At that, such events are supposed to happen in some area where the system has access to a fixed and the same set of environmental resources.

5 Conclusion

The possible knowledge fusion effects were described. These effects were found in the context-aware DSS for the emergency response domain. Knowledge fusion patterns specifying these effects along with the states of knowledge sources involved in the knowledge fusion processes were revealed. One of the possible pattern applications, which is the pattern-based methodology for design of context-aware DSSs, was offered. In the methodology the patterns were used to specify the system requirements ensuring the full system functionality.

So far, the methodology covers the executive phase of the DSS. Some future research is needed to specify the overall set of the system requirements and to describe information/knowledge flows across the DSS' stages and between the patterns. Such a specification will enable to find out dependencies between the system stages and available functionalities, and to specify more accurately the applicability of the knowledge fusion patterns to different user's requirements.

Acknowledgments. The research was supported partly by projects funded by grants 11-07-00045, 11-07-00058, 12-07-00298 of the Russian Foundation for Basic Research, the project 213 of the research program "Information, control, and intelligent technologies & systems" of the Russian Academy of Sciences (RAS), and the project 2.2 of the Nano- & Information Technologies Branch of RAS.

References

1. Sage, A.P.: Decision Support Systems Engineering. John Wiley & Sons, New York (1991)
2. Vessey, I., Conger, S.: Learning to Specify Information Requirements: The Relationship between Application and Methodology. JMIS, 10 (2) 177–201 (1993)
3. Elliott G., Strachan, J.: Global Business Information Technology: an Integrated Systems Approach. Addison Wesley, Harlow (2004)
4. MDA – the Architecture of Choice for a Changing World, OMG Model Driven Architecture, http://www.omg.org/mda/ (access date: 05.07.2013)
5. Shahbazian, E., Pigeon, L., Krtaft, J., Bossé, É.: Building Technology-Enabled Decision Support Applications. In: Bossé, É., Shahbazian, E., Rogova G. (eds.) Prediction and Recognition of Privacy Efforts Using Collaborative Human-Centric Information Systems. IOS Press, Amsterdam, pp. 98–108 (2013)
6. Smirnov, A., Pashkin, M., Chilov, N. Levashova T., Haritatos, F.: Knowledge Source Network Configuration Approach to Knowledge Logistics. Int. J. Gen. Syst. 32 (3) 251–269 (2003)
7. Smirnov, A., Levashova T., Shilov, N.: Context-Based Knowledge Fusion Patterns in Decision Support System for Emergency Response. In: Comes, T., Fiedrich, F., Fortier S., Geldermann J., Műller, T. (eds.) 10th International ISCRAM Conference, Baden-Baden, Germany, pp. 597–606 (2013)
8. Alun, P., Hui, K., Gray, A., Marti, P., Bench-Capon, T., Cui, Z., Jones, D.: Kraft: an Agent Architecture for Knowledge Fusion. Int. J. Coop. Inf. Syst. 10 (1–2) 171–195 (2001)
9. Scherl R., Ulery D.L.: Technologies for Army Knowledge Fusion. Final report no. ARL-TR-3279, Computer Science Department, Monmouth University, Monmouth (2004)
10. Holsapple C.W., Whinston A.B.: Building Blocks for Decision Support Systems. In: Ariav G., Clifford J. (eds.) New Directions for Database Systems. Ablex Publishing Corp., Norwood, pp. 66–86 (1986)
11. Lee, K.-R.: Patterns and Processes of Contemporary Technology Fusion: The Case of Intelligent Robots. Asian J. Technol. Innov. 15 (2) 45–65 (2007)
12. Grebla, H.A., Cenan C.O., Stanca, L.: Knowledge Fusion in Academic Networks. BRAIN: Broad Res. Artif. Intell. Neurosci. 1 (2) http://www.edusoft.ro/brain/index.php/brain/article/download/60/145 (2010)
13. Laskey K.B., Costa P., Janssen T.: Probabilistic Ontologies for Knowledge Fusion. In: 11th IEEE International Conference on Information Fusion, http://ieeexplore.ieee.org/xpl/-freeabs_all.jsp?arnumber=4632375 (2008)
14. Jonquet, C., LePendu, P., Falconer, S., Coulet, A., Noy, N.F., Musen, M.A., Shah, N.H.: NCBO Resource Index: Ontology-Based Search and Mining of Biomedical Resources. J. Web Semant. 9 (3) 316–324 (2011)
15. Kuo, T.-T., Tseng, S.-S., Lin, Y.-T.: Ontology-Based Knowledge Fusion Framework Using Graph Partitioning. In: Chung, P.W.H., Hinde, C.J., Ali M. (eds.) IEA/AIE 2003. LNAI, vol. 2718, pp. 11–20 (2003)
16. Gou J., Yang J., Chen Q.: Evolution and Evaluation in Knowledge Fusion System. In: Mira J., Alvarez, J.R. (eds.) IWINAC 2005. LNCS, vol. 3562, pp. 192–201 (2005)
17. Roemer, M.J., Kacprzynski, G.J., Orsagh, R.F.: Assessment of Data and Knowledge Fusion Strategies for Prognostics and Health Management. In: 2001 IEEE Aerospace Conference, vol. 6, pp. 2979–2988 (2001)
18. Lin, L.Y., Lo, Y.J.: Knowledge Creation and Cooperation Between Cross-Nation R&D Institutes. Int. J. Electron. Bus. Manag. 8 (1) 9–19 (2010)