UDK 658.512

ABOUT THE THEORETICAL FOUNDATION FOR A META-CONTEXT INTERCHANGE TECHNOLOGY

Z. V. Dudar[1], O. G. Vorochek[2]

[1] KhNURE, Kharkiv, Ukraine, dudar@kture.kharkov.ua;

[2] KhNURE, Kharkiv, Ukraine, relf@kture.kharkov.ua

The proposed research is devoted to investigation Context Interchange technology [1] as a new approach for achieving semantic interoperability among autonomous and heterogeneous databases. This technology is intended for the organization of logical connectivity among sources and consumers of large-scale semantically heterogeneous systems. The Context Interchange technology assumes comparison of contexts associated with any data source and consumer of semantically heterogeneous databases and elimination of semantic conflicts that are detected [2]. This comparison uses ontologies as specifications of concepts, their properties and relationships between them in a knowledge domain. This work is concentrated on building, integrating and using ontologies in the Context Interchange technology.

ONTOLOGY, CONTEXT, INTEROBERABILITY, ONTOLOGICAL GRAPH, META-CONTEXT INTERCHANGE TECHNOLOGY.

Introduction

The wide proliferation of Internet and Web-based technologies has caused a large amount of data that is often stored in databases with fundamentally different structures and semantics. Attempts to integrate such data from the diverse databases frequently cause semantic conflicts that inhibit data exchange. There are some approaches for resolving these conflicts [1]. One of the new approaches for integrating large-scale semantically heterogeneous databases is Context Interchange [1, 2]. These works are devoted to development of Context Interchange technology that is intended for organizing logical connectivity among sources and consumers of the diverse databases. Logical connectivity means ability to exchange meaningful information. The main component of this technology is the Context Mediator, which is an intelligent agent that performs data conversions between the source and consumer. It performs the conversions comparing contexts associated with any two databases involved in data exchange and eliminating semantic conflicts that are detected [2]. The Context Interchange technology is implemented in the Context Interchange system. The top-level structure of the Context Interchange system is developed [1,2].

The preceding works [10,11] to this work are devoted to investigation and adaptation of multilevel knowledge representation models for formalizing the semantics of the data in semantically heterogeneous database, and implementation of these models in the Context Interchange technology. Namely, these works were devoted to modeling multilevel database contexts and their comparison for elimination of semantic conflicts.

The comparison of contexts, detection and elimination of semantic conflicts in the Context Interchange technology is performed by the Context Mediator using Common Ontology, a context-independent area in the knowledge base of the Context Mediator. The Common Ontology is a formal specification of concepts and relationships between them, which are common (or understood) for any source or consumer involved in data exchange. Gruber [3] defines ontology as «an explicit specification of a conceptualization». The problem of applying the Context Interchange technology to large-scale databases is the components of such databases can belong to different domains of knowledge. That is why it is necessary to integrate domain ontologies of the systems involved in data exchange with the goal to build the area of Common Ontology. This work is devoted to the integration of ontologies as the process of building new ontologies from the existing in terms of some methodology.

The definitions and basic principles and criteria for the design of ontologies are introduced in [3]. The classification of ontologies is described in [5, 6]. Within these classifications and the task of context mediation the most interesting types of ontologies are: terminological ontologies to specify the terms that are used to represent knowledges in a domain, information ontologies that specify the record structure of databases, knowledge modeling ontologies that specify conceptualization of the knowledge, and domain ontologies to express conceptualizations that are specific for particular domains [5]. Some existing methodologies for bulding ontologies are introduced in [4, 7, 8, 9]. Within the task of building Common Ontology for the context mediation the most interesting are methodologies by Uschold and King [7, 8], and by Gruninger and Fox [8, 9]. The Skeletal Methodology by Uschold and King provides general guidelines for developing ontologies. This methodology consists of four stages: identification of purpose, building the ontology, evaluation and documentation. The stage 2 of this methodology is very reasonable for building the Common Ontology for the Context Interchange technology because it fits in the previously used methodolo-

gy for building contexts of databases [10,11]. The methodology by Gruninger and Fox involves building a logical model of the knowledge. It assumes informal description of specifications with its further formalization. This methodology is also considered for building the Common Ontology as a formal approach to ontology design.

In terms of this the Skeletal Methodology the ontology to be built is the Common Ontology for the Context Interchange technology. The process of building the Common Ontology assumes three steps [6]: ontology capture, coding ontology using a formal language, integrating existing ontologies. This work is concentrated on the second and third steps. The ontological graph is proposed for formalizing ontologies, the genetic search method is proposed for providing integration of ontologies.

Because the role of ontologies is to be base for «understanding» data contexts, the Context Mediator, where process of integrating ontologies is implemented, becomes Meta-Context Mediator. In this case the process of data conversion between the source and consumer, which involves integrating ontologies, is called meta-context mediation [12].

1. Formalizing ontology

The basic knowledge representation model for formalizing ontology in this work is *ontological graph* (*ontograph*). Formally, ontograph is the following:

$$G_o = (V_o, E_o), \tag{1}$$

where V_o – a set of nodes formalizing concepts of the ontology; E_o – a set of arcs formalizing relationships between the concepts.

The following adjacency matrix is used for the description of the ontograph:

$$A = (a_{ij}), \text{ dimension } |V_o| * |V_o|, \tag{2}$$

$$a_{ij} = \begin{cases} S_{Eoij} \times K_{doij}, \ if \ (i,j) \in E_o \\ 0, \ \text{otherwise}, \end{cases} \tag{3}$$

where S_{Eoij} – semantic of a relationship, K_{doij} – data quality coefficient.

The ontological graph is multilevel, because during the acquisition and forming ontological knowledge the installation of steady associations is happening. A steady association is a set of ontological concepts and relationships among them unambiguously describing some knowledge. The steady association is considered as a new concept on a higher level of the ontograph. Because the ontograph is multilevel it is necessary to link its levels. The following transition matrix is used for formalizing the transitions between levels:

$$P = (p_{lk}), \text{ dimension } |\mathscr{L}_{S_o}| * |V_o'|, \tag{4}$$

$$p_{lk} = \begin{cases} 1, \ if \ (l,k) = \text{true} \\ 0, \ \text{otherwise}, \end{cases} \tag{5}$$

where $\mathscr{L}_{S_o}$ – set of steady associations; V_o' – set of associated concepts.

A new adjacency matrix can describe the steady associations:

$$As = (As_{mn}), \text{ dimension } |\mathscr{L}_{S_o}| \times |\mathscr{L}_{S_o}|, \tag{6}$$

$$As_{mn} = \begin{cases} a_{ij}, \ if \ (i,j) = \text{true} \\ 0, \ \text{otherwise}. \end{cases} \tag{7}$$

The task of searching semantics of concepts is laid in the searching chains of semantic links between the corresponding concepts of integrated ontologies.

2. Principle of forming context-meaningful ontological chains

A context-meaningful ontological Chain (CMOC) – is a sequence of nodes and arcs of ontological graph, which meaningfully describes properties of an on object in terms of the import and export contexts. There are 2 types of CMOC depending on use: static CMOC is used if the import and export contexts are constant and can be reused for a long time; dynamic CMOC is used if the import and export contexts are variable and belongs to different (but adjacent) domains. For achieving interoperability in the static case the search of a semantic link between any two objects is leading using methods and algorithms of the graph theory for the searching of the shortest path. In the case of dynamic CMOC the goal is finding the shortest path for a limited period of time while this path is the most qualified, that is why the only classic methods of search are not appropriate. A genetic search method is proposed in this work. In terms of the meta-context interoperability the main task is the search of CMOC that minimize a number of context conversions between the import and export contexts.

A scheme (fig. 2) represents semantic links in case of meta-context mediation for integration of two databases. There are the following possible ways of the process of integration:

1) DB1 – Import Context – Mediator – Export Context – DB2;

2) DB1 – Ontology1 – Mediator – Ontology2 – DB2;

3) DB1 – Import Context – Ontology1 – Import Context – Mediator – Export Context – Ontology2 – Export Context – DB2;

4) DB1 – Ontology1 – Import Context - Ontology1 – Mediator – Ontology2 – Export Context – Ontology2 – DB2.

The first case does not consider ontological knowledge about domains of the databases, the second – context. That is why these schemes are not appropriate. Both #3 and #4 ways are reasonable, but in the case #4 context conversions are necessary the only to form

context representation of data, and the rest of integration process is based on fuzzy ontological links between domain ontologies. However this way requires defining context-ontological relationships and integrating ontologies. These two steps are described below.

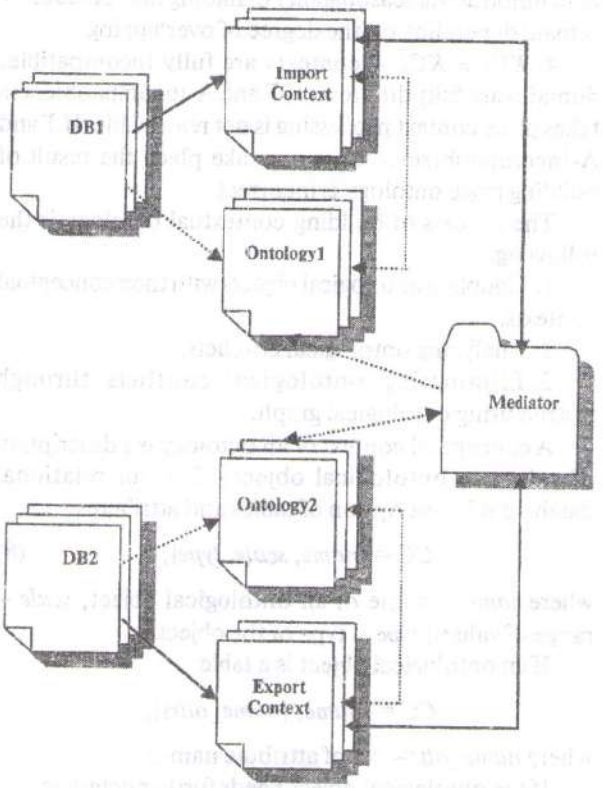

Fig. 1. Data flows in meta-context mediation

3. Forming context-ontological relationships in the knowledge base of the Meta-Context Mediator

In terns of the meta-context mediation domain ontology – is a set of object and relationships between them, which are described by conceptual model of a database. Context describes relationships between particular data and its semantics (concepts) in the particular databases. There are the following context and ontological concepts useful for the meta-context approach:

- basic ontology (BO) – is a basic ontological graph formed for any source or consumer;
- integrated ontology (IO) – is a result of integration of basic ontologies;
- steady associated ontology (AO) – is a resulting ontology with steady associations; this is the higher-level ontology;
- contextual ontology (CO) – is an ontology coupled with a context of any source of consumer involved in data exchange;
- key context (KC) – a set of key concepts and relationships between them for a domain of sources and consumers involved in data exchange;
- conceptual context (CC) - is defined for a fragment of a database (f.e. for only source's database, for a table, for an attribute).

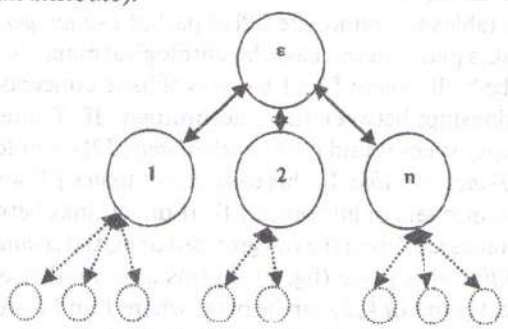

Fig. 2. Context hierarchy

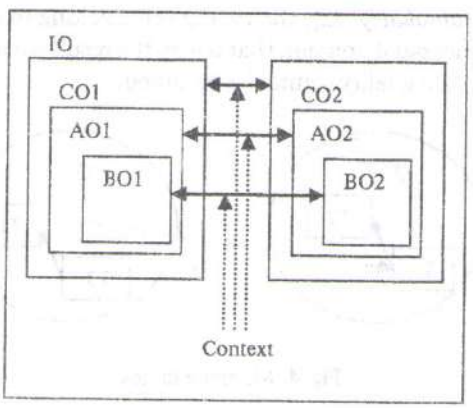

Fig. 3. Ontological hierarchy

Both contexts and ontologies are organized as hierarchy. In the hierarchy of contexts (fig. 2) root ε – is a key context describing an external representation of a domain, branches and leaves – are conceptual contexts describing detail properties of database fragments. Basing the hierarchy of ontologies (fig.4) it is possible to describe a process of forming an integrated ontology:

$$IO = CO + KC = CO1 + CO2 + \Sigma CC =$$
$$= AO1*CC1 + AO2*CC2 + \Sigma CC$$

Thus, for finding an integrated ontology it is necessary to form basic ontologies of systems involved in data exchange, to find steady associations and to resolve ontological conflicts between contextual ontologies.

4. Building basic ontologies

An integration of ontologies of two systems involved in data exchange requests the following steps:

1) ontology capture and forming ontology graph,

2) resolving ontological conflicts and genetic search of ontological chains for interoperability.

The first step is based on the mapping of ontology and data of the first system involved in data exchange into according ontology and data of the second system.

For example, for two relational databases, where sets of predefined ontologies O_1 and O_2 are identical to sets

of tables: $O1 \equiv \{T1\}$, $O2 \equiv \{T2\}$, it is necessary to map $\{T1\} \rightarrow \{T2\}$ and install links between tables.

If direct (or clear) links between semantically equivalent tables are found, the full or partial *T-interoperability* takes place. In this case the ontological graph $G(V,E)$ can be built, where V and E – sets of basic concepts and relationships between them accordingly. If *T*-interoperability is not found ($\{T1\} \neq absolutely\{T2\}$) ontologies are *T-incompatible*. In this case sets of tables $\{T\}$ are divided into sets of attributes $\{A\}$. If direct links between attributes are found the full grouped or partial *A-interoperability* takes place (fig. 7). In this case another ontological graph $G(V,E)$ can be built, where V and E are sets of attribute-level basic concepts and relationships between them. If A-interoperability is not found ($\{A1\} \neq absolutely\{A2\}$) the two systems belong to different conceptual domain, that is why the result wit appropriate data quality cannot be obtained.

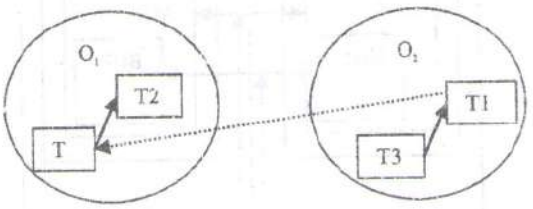

Fig. 4. Mapping tables

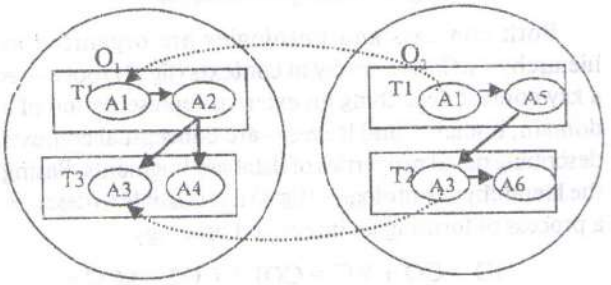

Fig. 5. Mapping attributes

5. Building contextual and steady associated ontologies

The process of building contextual ontology consists of finding hidden relationships, analyzing clear relationships and coupling them. Within this process key context (KC), which defines key concepts of a domain, can be a criteria for possibility of integrating potentially incompatible systems. The analysis of key contexts of two systems involved in data exchange (KC1 and KC2) is the first step of building contextual ontology. There are some types of relationships between two key contexts:

1. $KC1 \approx KC2$ – integrated objects belong to the same domain. All concepts of the first system are unambiguously mapped into concepts of the second system. Finding interoperability is easy, and assumes the only classic search of context-ontological links between objects.

2. $KC1 \in KC2$ – domains are nested. Finding interoperability is the same as in (1).

3. $KC1 \approx KC2$ – domains are partly overlapped. A conclusion about reasonability of finding interoperability is maid depending on the degree of overlapping.

4. $KC1 \neq KC2$ – contexts are fully incompatible, domains are fully different. If T and A-incompatibleness takes place context processing is not reasonable. If T and A-incompatibleness does not take place the result of building basic ontology is incorrect.

The process of building contextual ontology is the following:

1. Coupling ontological objects with their conceptual contexts.
2. Analyzing ontological conflicts.
3. Eliminating ontological conflicts through restructuring ontological graph.

A conceptual context of an ontology is a description of values of ontological objects. F.e. for relational database it is description of tables and attributes:

$$CC = \{name, scale, type\}, \quad (8)$$

where *name* – name of an ontological object; *scale* – range of values; *type* – type of the object.

If an ontological object is a table:

$$CC = \{name, \{name\_attr\}\},$$

where *name_attr* – set of attribute names.

If the ontological object needs further detailing:

$$CC = \{name, \{name\_attr, scale, type\}. \quad (9)$$

After contextual ontology is built it is necessary to resolve ontological conflicts to build steady associated ontology.

Assume two sets of objects are built: $O_1=\{o_{1i}\}$ – describes a set of ontological objects of the first system involved in data exchange, $O_2=\{o_{2j}\}$ – the second. Assume that relationships between objects are defined, and links for interoperability are found.

The first step of resolving ontological conflict is eliminating homonyms. Assume we have two potentially interoperable objects $o_{1i} \in O_1$ и $o_{2j} \in O_2$. If conceptual contexts $CC1 \equiv CC2$ the probability of homonymy is very low, that is why the link between these objects is correct. If $CC1 \neq CC2$ then homonymy exists.

F.e., we have two ontological fragments (fig. 6) with potential interoperability, equal conceptual contexts for the objects #3 and different conceptual contexts for the object #1. Thus we have the following set of conceptual contexts:

$$CC1_1 = \{name = 1, scale1, type1\}$$
$$CC1_2 = \{name = 2, scale2, type2\}$$
$$CC1_3 = \{name = 3, scale3, type3\}$$

$CC2_1 = \{name = 1, \{name\_attr1\}\}$

$CC2_2 = \{name = 4, \{name\_attr2\}\}$

$CC2_3 = \{name = 3, scale3, type3\}$

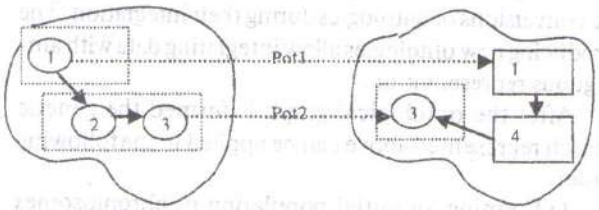

Fig. 6. Homonymy

There are two potential links:

Pot1: $O1_1 \text{ -> } O2_1$, $CC1_1 \neq CC2_1$

Pot1: $O1_3 \text{ -> } O2_3$, $CC1_3 = CC2_3$

In this case objects $O1_1$ and $O2_1$ have the same name but different conceptual contexts, that is why the link Pot1 is excluded and one of the objects is renamed to 1_*hom* (this renaming is applied only for ontological object but not for physical data structure). This renaming allows processing homonymic objects independently. The result of eliminating homonymy is represented on a figure 7.

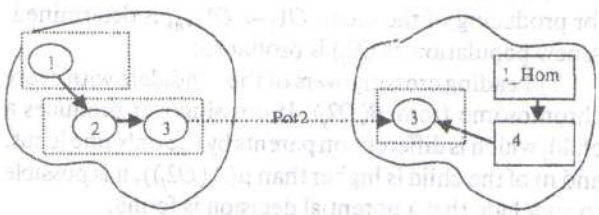

Fig. 7. Eliminated homonymy

Analysis of synonymy is more complicated because synonymy assumes full coincidence of conceptual contexts and links to other objects for two objects in different ontologies. The coincidence of conceptual contexts only assumes potential synonymy; that is why analyzing synonymy is leading by the following way:

1) Extraction of an integrated object of the first ontology;

2) Forming a set of links from this object to other objects. Thus we obtain a mask of interoperability;

3) Comparing this mask with every object of the second ontology. Coincidence means synonymy – this is ideal case because interoperability is achieved, or steady association is installed (fig. 8, object #2 and #6). Otherwise we have partial synonymy – it means ontological conflict exists (fig. 9) and should be eliminated.

In the case of partial synonymy (fig. 9) key object #5 of the second ontology (part of its link is overlapped by the mask) is split on two objects: with overlapped (#5) and not overlapped ($\#5_{link}$) links. Thus, a quasi-steady association is installed between the integrated objects #2 of the first ontology and the object #5 of the second one. A link of "neighborhood" is installed between objects #5 and $\#5_{link}$. The next step is search meaningful ontological chains between integrated ontology for achieving interoperability.

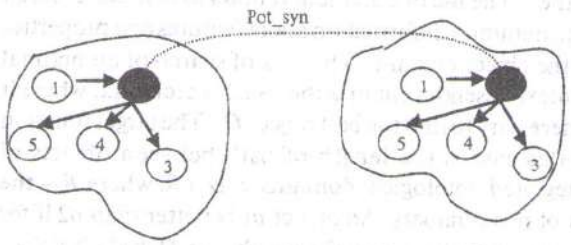

Fig. 8. Synonymy

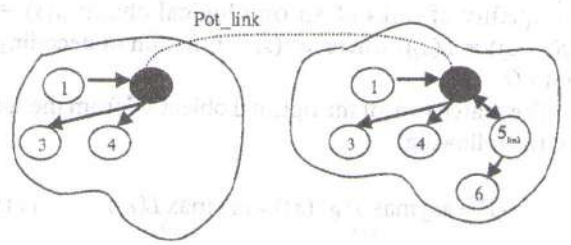

Fig. 9. Quasy-steady link

6. Search of ontological chains for achieving interoperability

After installation of steady associations, basing on these sets of ontologies, it is necessary to search ontological chains for interoperability. If integrated objects are compatible links between them are clear and interoperability is guaranteed. If integrated objects are incompatible it is necessary to use indirect methods to provide interoperability. In this case the task of searching compatibility between two objects $O1_i \rightarrow O2_j$ is the task of searching an ontological chain $O1_i \rightarrow O1_{i+n}$, where $O1_{i+n} \equiv O2_{j+n}$, and the length of the chain $L(O1_i \rightarrow O1_{i+n})$ must provide quality as closer as possible to the required quality of the result. This length of chain is assumed as minimum length for integration of the objects. In this case the following equation should be true:

$$L(O2_{j+n} \rightarrow O2_j) \geq Ll(O1_i \rightarrow O1_{i+n}), \quad (10)$$

where $O2_{j+n} \equiv O1_{i+n}$ and $O2_j \approx O1_i$.

The second model, which is used by the meta-context mediator, is intended for the genetic search of context-meaningful ontological chains for interoperability. The goal of the search of the optimal context-meaningful chain for the first system involved in data exchange is the finding ontology of the second system participating in data exchange. This ontology should correspond to

ontology of the fist system with the predetermined probability. There are two domains in this model: domain of ontological objects O and domain of statements S, where: $O \neq S$.

Representation is mapping from the O to the S. In our case it is encoding function $e: O \rightarrow S$.

For any object o from the O and a statement s from the S the formula $s=e(o)$ means that the s is the statement of the o. The use of statements allows us to make a search with minimum information about features and properties of the object domain. The task of search of an optimal context-essential chain is the usual search task, where it is necessary to find the best object O^*. The target function in this case is the length of path between objects of integrated ontological domains: $f: O \rightarrow R$, where R – the set of real numbers. An object $o1$ is better than $o2$ if for any two objects $o1$ and $o2$ from the set $O: f(o1) > f(o2)$.

The fitness-function is necessary for evaluation of statements:m: $S \rightarrow R$, where R – the set of real numbers, μ — quality of links of an ontological chain: $\mu(s) = f(e^{-1}(s)) = L(o_i)$, where $e^{-1}(s)$ – function of decoding S into O.

The search task of the optimal object O^* from the set O is the following:

$$O^* = \arg\max_{s \in S} f(e^{-1}(s)) = \arg\max_{o_i \in O} L(o_i). \qquad (11)$$

The resolving this task is happening through the search of the optimal representation s^* in the domain S:

$$s^* = \arg\max_{s \in S} \mu(s). \qquad (12)$$

The adjacency matrixes, which describe ontology, are used in the proposed genetic search. These matrixes consider steady associations. All elements of an adjacency matrix, which describes mutual correspondences between objects, must be represented as single directed correspondence with the computation of data quality indicator. According to a query the sub-domain of statements from the adjacency matrix is defined.

The obtained domain has the $n \times m$ dimension, where n – the total number of ontological objects, m – the number of ontological objects linked to the object O^*.

Use of the genetic model in the meta-context mediation supposes redefinition of some classical concept of this formalism.

Mutation operator changes the particular selected hen with the value 1 rather than random hen. The mutated hen is changed to 0 and determines a collection of locus (considering the object O), which state is changed to 1. This mechanism allows considering possible links among the objects of ontology.

Inversion destroys the sequence of fragments of chromosome from parent to child. It means inclusion or exclusion of a possible property of ontology.

Crossingover produces a new chromosome with the properties from the both of parents. It allows changing a recessive fragment of chromosome to a dominant one that can give a better result of the search.

The proposed genetic approach to the knowledge modeling allows resolving a problem of modeling semantic conversions of ontologies during their integration. The producing new ontologies allow integrating data with ambiguous representation.

After the ontological graph is formed the genetic search represented above can be applied in the following order:

1) Forming an initial population of chromosomes $B(O1_i)=(A_1(O1_i) \ldots A_n(O1_i))$;

2) Finding hens, which are set to 1 and their locus correspond to the integrated object;

3) Mutation of the $B(O1_i)$ and finding a hen of interoperability for the new population $B'(O1_i)=(A_1'(O1_i) \ldots A_{i-1}'(O1_i), A_{i+1}'(O1_i), \ldots A_n'(O1_i))$;

4) Repeat 2 and 3 while $B^{(m)}(O1_i) \neq (0)$ or while it is reasonable. A chromosome can be included into the population of candidates only if $\mu(A_{i+1}(O1_i)) \geq \mu(A_i(O1_i))$;

5) Selecting a chromosome from the population $B_\kappa(O1_i)$ with the maximum $\mu(A_i(O1_i))$. The hen of interoperability of this chromosome defines an object $O1_{i+n} \approx O2_{j+n}$.

6) The number of mutations N_i, which are necessary for producing of the chain $O1_i \rightarrow O1_{i+n}$, is determined. A new population $B(O2_j)$ is produced;

7) Leading crossingovers of the candidate with every chromosome from $B(O2_j)$. If crossingover produces a child, which is different on parents by the only one locus, and m of the child is higher than $\mu(A_j(O2_j))$, it is possible to conclude that a potential decision is found;

8) The produced child, the parent are decoded.

Thus a fragment of ontological chain is found. Iterative call of this algorithm allows finding almost all ontological chains for providing interoperability for integration of ontologies.

Conclusion

This paper describes two models, which are useful for integration of ontologies in the Context Interchange technology. The first model is ontological graph, which is used for modeling ontologies. The second model is genetic search of ontological chains on the ontological graph, which is intended to find interoperability between integrated ontologies. The integration of ontologies is targeted to form the area of Common Ontology, which is used in the Context Interchange technology as the base for comparison of contexts and elimination of semantic conflicts.

Literature: 1. *Goh, C. H., Madnick, S. E., and Siegel, M. D.* Context interchange: overcoming the challenges of large scale interoperable database systems in a dynamic environment. In

Proceedings of the Third International Conference on Information and Knowledge Management (Gaithersburg, MD, Nov 29—Dec 1 1994), pp. 337—346. 2. *C. Goh*, «Representing and reasoning about semantic conflicts in heterogeneous information systems», Ph.D. Thesis, MIT Sloan School of Management, 1996. 3. *Gruber T.*, (1995), Towards Principles for the Design of Ontologies Used for Knowledge Sharing, International Journal of Human-Computer studies, 43 (5/6): 907 — 928. 4. *M. Uschold and M. Gruninger.* Ontologies: principles, methods and applications. The Knowledge Engineering Review, 11(2):93—136, November 1996. 5. *G. van Heijst, A. T. Schreiber, and B. J. Wielinga.* Using explicit ontologies in KBS development// International Journal of Human-Computer Studies, 46(2/3).- 1997.- 183—292 6. *Guarino N.* Some ontological principles for designing upper level lexical resources. In Proceedings of the First International Conference on Language Resources and Evaluation, Granada, 1998. 7. *M. Fernandez Lopez,* Overview of Methodologies for Building Ontologies. Workshop on Ontologies and Problem-Solving Methods: Lessons Learned and Future Trends, IJCAI-99. Stockholm, Sweden. August, 1999. 8. *M. Uschold and M. King.* Towards a methodology for building ontologies. In Proceedings of the IJCAI-95 Workshop on Basic Ontological Issues in Knowledge Sharing, Montreal, Canada, 1995.9. *M. Gruninger and M. Fox.* Methodology for the design and evaluation of ontologies. In Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing held in conjunction with IJCAI-95, Montreal, Canada, 1995. 10. *Ye. V. Biletskiy, O. O. Biletska, M. F. Bondarenko and Z. V. Dudar.* The Context Interchange Technology Based on the Predicate Knowledge Representation Model // The 2002 International Conference on Artificial Intelligence (IC-AI'02: Las Vegas, USA, June 24-27, 2002). 11. *M.F. Bondarenko, Z.V. Dudar, Ye.V. Biletskiy, O.G. Baygozina.* The Context Interchange System Framework // Problems of Bionic (Problemi Bioniki), KTURE, Kharkiv, 1999, No 50. P.18-24. 12. *Baygozina O.G.* «Using ontology in the context mediator system»// 6-th International Conference "Theory and technique of information receiving, handling and sending", Kharkiv, 2000.

Поступила до редколегії 15.03.07