Intelligent search in Big Data

E Birialtsev, N Bukharaev and A Gusenkov

1 R&D Director Gradient Ltd, PhD, Kazan, Russia
2 Associate Professor, PhD, Institute of Computational Mathematics and Information Technologies, Kazan Federal University, Kazan, Russia
3 Senior Lecturer, PhD, Institute of Computational Mathematics and Information Technologies, Kazan Federal University, Kazan, Russia

E-mail: 1igenbir@yandex.ru, 2boukhary@gmail.com, 3gusenkov.a.m@gmail.com

Abstract. An approach to data integration, aimed on the ontology-based intelligent search in Big Data, is considered in the case when information objects are represented in the form of relational databases (RDB), structurally marked by their schemes. The source of information for constructing an ontology and, later on, the organization of the search are texts in natural language, treated as semi-structured data. For the RDBs, these are comments on the names of tables and their attributes. Formal definition of RDBs integration model in terms of ontologies is given. Within framework of the model universal RDB representation ontology, oil production subject domain ontology and linguistic thesaurus of subject domain language are built. Technique of automatic SQL queries generation for subject domain specialists is proposed. On the base of it, information system for TATNEFT oil-producing company RDBs was implemented. Exploitation of the system showed good relevance with majority of queries.

1. Introduction
Exponential growth of input data volume, measured today in zettabytes, led to qualitative changes in computer aided technologies of acquiring, storing, processing and analysing information known as the Big Data phenomena. Forrester Research Company [1], for instance, defines the notion as a hardware or software technology to organize, control and analyse data arrays, characterized by “four Vs” — Volume, Variety, Variability and Velocity. NoSQL [2], MapReduce [3], Hadoop [4], SAP HANA [5] are the most widely known technologies of the kind.

Availability of sufficiently effective search procedure can be stated as one of the main goals for data integration. On the other hand, volume and variety of the Big Data makes problem of integration of heterogeneous information resources non-trivial. M. Kogalovsky proposed in [6] the classification of data integration systems, based on division of physical, logical and semantic levels of integration. Integration on physical level means plain conversion of information taken from heterogeneous data sources to some common format of physical representation. Integration on logical level provides a global scheme of uniform access to such sources, describing common data representation with respect to structural and (in object-oriented models) behaviouristic properties of the data sources. Integration on semantic level supports data representation reflecting just semantic properties of information, described within framework of some subject domain ontology. The latter allows to use ontology terms as conceptual reasoning model, giving great advantage to base end user interface on high level data model. For ontology description OWL language, developed by Semantic Web Activity working group and recommended by W3C consortium [7], is usually used.
2. RDBs integration
Let’s describe, in its most essential features, the following approach for RDB integration, aimed on organization of effective search in large information systems, containing some several dozens of local RDBs. Those RDBs may have different logical structure and physical representation, but are related to one subject domain. For such integration, we have to use structural information, retrieved from the databases, and more specific information on the subject domain. In other words, to solve the problem of data integration on semantic level we need to define explicitly auxiliary information resources such as physical RDBs model, logical model of subject domain and thesaurus of end user terminology, presented in the formalism of ontologies.

3. RDBs ontology
For RDBs integration we need to retrieve information about database structure. Structural problems appearing here are thoroughly analysed in [8–10]. To output meaningful information, the search engine should have information about possible table joins. Such information can be easily retrieved from database schemes, as modern RDBs store system information on keys. Note also that majority of the modern RDBs store information on possible joins to support data integrity. So, restoring all possible meaningful combinations of joining RDB tables is a fairly simple task.

There are many well developed formalisms to describe database structure, such as DDL SQL, ER-diagrams [11], UML [12] etc. Their expressive power is sufficient for solving problems of database design and data access support in the case when each database is considered separately, each one with its own system of attributes. But let’s note that in the case of RDBs data integration, there arise problems, related to the peculiarities of naming of database artefacts (such as tables and fields) [13]. These peculiarities force us to treat the artefacts names not just as atomic units, but as independent semantically linked objects with their own structure.

There are several ways to describe RDB structure in terms of ontologies, which can be reduced to the following two main approaches.

The first one [14] implies
• conversion of the set of database tables to the set of ontology concepts, having the same name and slots, corresponding to the tables columns, and
• projection of joins of tables into relations between concepts in the form of migrating keys.

The set of data domains, i.e. all predefined and user defined data types, is also included into set of ontology concept. Thus, in this case table records, key values and data types form the set of instances of ontology concepts.

We prefer, as more universal one, the second approach proposed in [9, 10], which presumes higher level of the concepts abstraction. In this case RDB structure is described in terms of universal concepts such as
• TABLE, COLUMN, KEY, DOMAIN, corresponding to the main RDB objects, and
• General relations between the objects such as
  o TABLE contains COLUMN,
  o TABLE has primary KEY,
  o TABLE has foreign KEY,
  o COLUMN contains KEY and
  o COLUMN has type DOMAIN.

Two following interpretation functions are also introduced [15]:
φ1: If TABLE1 has primary KEY1 and TABLE2 has foreign KEY1, then there exists TABLE3, containing all columns of TABLE1 and TABLE2.
φ2: If TABLE1 contains COLUMN1, then there exists TABLE2, containing all columns from TABLE1 except COLUMN1.

The first function corresponds to the table join operation, and the second one corresponds to the relation projection operation. The second function is used to select desired columns from table joins. The data integration problem then is reduced to the problem of finding different ways to extract
the specified attributes (columns) from the tables of the RDBs — i.e. finding such sequence of \( \phi_1 \) and \( \phi_2 \) functions applications, that gives in the end desired set of columns from the given ontology. This problem belongs to the well-known class of problems on a walking along the oriented graph, in which vertices represent instances of TABLE concept and edges reflect presence of the one-to-many relation. Methods of solving this class of problems are well known [16], although increase of computation complexity depending the problem dimension should be taken onto account.

4. **Subject domain ontology**

Creating ontology is a time-consuming process, which usually is carried out by a team of specialists, highly qualified in computer linguistics and in specific subject domain. Let’s describe here technique proposed in [9, 10, 17, 18] for oil-producing ontology creation.

Oil industry standard EPICENTRE logical model, developed by Petrotechnical Open Software Corporation (POSC) [19], was chosen as an initial ontology prototype. The model is represented in the form of the set of ER-diagrams considered together with the set of text files on EXPRESS object-oriented language. It defines more than 1000 real technical and business objects (or entities, in the POSC terminology), related to the field of oil exploration and production.

As usual in the object-oriented approach, the objects here are characterized by their attributes. The most important objects attributes define relationships between entities. Object-oriented concept of inheritance is also an essential part of the EPICENTRE architecture, providing effective way of organizing extremely large set of entities into a compact logically linked structure.

In our approach OWL-DL dialect of the OWL language [7], developed by Semantic Web Activity working group and recommended by W3C consortium, has been chosen as a unified ontologies representation language. The core of the approach is the scheme of conversion of the EPICENTRE model to its OWL-DL description. To implement the scheme, LR(1) formal grammar of the EPICENTRE model was defined. The Russification of the description of the EPICENTRE entities and attributes, as well as all corresponding classes and properties on OWL, was also performed.

5. **Linguistic thesaurus**

Creation of the linguistic ontology of natural and technical objects follow the main principles of the WordNet thesauri construction [20]. The dictionary of the subject domain is constructed by combining word phrases from descriptions of entities and attributes of the EPICENTRE model with word phrases taken from descriptions of tables and domains attributes of the TATNEFT oil-producing corporation RDBs. The lexical and semantic characteristics of the databases, used in constructing the dictionary, are described in detail in [9, 21].

For each word phrase an input set of synonyms or synset is defined. On the set of lexical-semantic variants of word phrases and synsets, the following linguistic relationships are defined: hyponymy, part-whole, incompatibility, antonymy, converseness, homonymy. The synsets of the thesaurus constructed in this way inherit the features specific to real descriptions of the RDB attributes, such as the presence of short phrases, general and technical abbreviations, spelling errors. Note however that, unlike the analysis of texts in natural language, it is possible to refine the recognition of input words by viewing the contents of the tables attribute descriptions and comparing them with the subject domain ontology.

6. **Natural language search and SQL queries generation**

In this section, we describe in brief a way of automatic generation of SQL queries, which requires from the end users to be knowledgeable just in the subject domain. This technique uses subject domain ontology, instance of RDB ontology and thesaurus of user terminology described earlier. At first, we should stress here that exactly the automation of the subject domain ontology construction, the linguistic thesaurus and universality of the RDB ontology makes described approach usable by Big Data collection, storage and processing technologies [1, 2]. Note also that automation of the mentioned objects construction allows us to create information systems more resistant.
to modifications in process of usage. In case when the logical model of the subject domain is changed, the subject domain ontology can be reconstructed and the linguistic thesaurus can be replenished quite easily.

From existing implementations, let’s point to the InBase system, developed by the school of A. S. Narinyani [22] as the closest to the proposed approach. A distinctive feature of the InBase is usage of advancing semantic analysis for parsing and understanding queries. The analysis here is based on the object model of the subject domain, linked by the designer to the RDB model. This approach has a number of significant advantages in comparison with the syntactic templates usage. In particular, it eliminates the need to specify in the query patterns all word forms and all possible orders of words.

At the same time, it should be recognized that the InBase system has not found wide application yet. As it seems to us, the approach developed in InBase system has a number of essential problems. The most significant of them is the interpretation of RDB columns as attributes of classes of the object model. Real databases quite often implicitly refer to the complex notions, that can be hardly adequately mapped to the object model — since they rather mean a subquery rather than a classic attribute. This issue has been considered in detail in [13, 21], so let’s here confine ourselves just to a single counterexample. Say, in one of the explored real industrial RDBs there were columns with comments like "The well production rate before repair" and "The well production rate after repair", which cannot be described as class attributes.

The second problem is necessity to bind the columns to the object model attributes manually. In addition to the great complexity of this process during the initial implementation, it also creates difficulties during system maintenance. In fairly often cases of changing RDB structure, labor intensive adjustments of the object model and its projection to the tables columns are required.

In corporate information systems, RDBs can contain hundreds of tables and thousands of columns. The investigation of the question which specific database, table and columns contains the required information can take several days, whereas the formulation of the query in natural language by the users knowledgeable in the subject domain will take several minutes. Note also that in such cases no specialist uses the natural language in all its richness, but is restricted to some simplified professional dialect.

In the proposed approach, information processing is based on the following principles:

- Dialogue between the system and the user is carried out in a table form;
- Semantic approach to columns search is used;
- Search of possible table joins by database keys is automated;
- Visual procedures for defining table selection operations are used.

The end user defines queries in the form of a table in natural language, using the terms of the thesaurus (subject domain language). See table 1 for an example of such query. Note that the query, formulated as a sentence in natural language, would be very cumbersome not only for computer analysis, but also for human understanding.

**Table 1. Query example**

| Column Name                  | Condition          |
|------------------------------|--------------------|
| Well No.                     | 10                 |
| Date of commissioning        | > 1.06.2001        |
| Date of overhaul             |                    |
| Expected oil production rate | > 0                |
| Actual oil production rate   |                    |
Recall that semantic information is presented in the system as a semantic network (ontology), containing a set of possible links between the concepts. Semantic parsing of the RDB columns comments is an almost completely automatic operation, which allows to quickly attach a large corporate database to the subject domain ontology. Semantic analysis of the user’s query is reduced to identifying the subgraphs of the network, which connect the concepts, corresponding to words phrases, used in the columns names.

Algorithm of semantic analysis of the user's query is revealing the subgraphs in the taxonomic tree of the subject domain ontology, connecting the concepts named by word phrases from the linguistic thesaurus, corresponding to the query columns (see figure 1).

**Figure 1.** Understanding of query in ontology context

7. Implementation of the system for Tatneft

In this section, we briefly outline the functionality of the web-based search system implemented in accordance with the approach described above for Tatneft oil-producing company. The system is described in more detail in [9, 10, 23, 24].

For user administration, a standard password entry scheme for individual users is used here. During the login process, each of the users is assigned one of three roles — User, Technologist or Administrator.

7.1. End user subsystem

Using the system's tools, the user constructs a query in the form of a table (see example in table 1). The names of the columns are formulated in a dialogue with the system in an arbitrary form with the use of phrases from the linguistic thesaurus. The number of columns is unlimited.

After setting the query, the user sends it to the generation. The system performs a semantic analysis of the query, builds the SQL expression and evaluates the query execution time to inform the user. The built-in SQL query is presented to the user, who can edit or run it for execution. The user interface is simplified as much as possible to allow any user with an understanding of the subject domain to work with the system. Knowledge of SQL language and database structures are not required for the average user.

For most queries, the system has shown good relevance of the results.
7.2. Technologist subsystem
This subsystem for flexible configuration of the system is of particular interest for evaluating the advantages of the approach. The main goal of a technologist is to mark up the columns of the database in a way that ensure the maximum relevance of the end user queries. To accomplish this task, he can use the following built-in tools:

- Subject domain language (thesaurus) editor;
- Subject domain ontology editor;
- Database ontology editor;
- Search algorithm configuration tools;
- Tools for analysis of completeness and adequacy of ontologies.

The system provides the built-in lexical editor to allow technologists to introduce new terms into the thesaurus. Note here that the presence in the user’s query of words absent in the thesaurus does not cause fatal consequences — an unknown word is marked, but it is ignored in the future analysis.

The ontology editor of the domain allows technologist to edit a specific concept or define a new one as well as to specify, delete or change the links of this concept with other concepts. Each link can be customized individually by changing the semantic distance between the concepts. This tool is also used to fine-tune the search engine at the stage of marking the names of user-defined columns and their matching with comments.

Technologists have an access to the list of all RDB tables, together with service information on the current state of their markup by concepts of the subject domain ontology. The subsystem allows them to specify the settings for the databases with which the system can work simultaneously. Thus, the subsystem allows to fulfil all necessary actions for tuning the system when determining the needed RDBs or changing the subject domain terminology are necessary.

7.3. Administration subsystem
The subsystem provides implementation of standard administrative functions such as maintaining the user registry, determining the access rights to work with system objects, logging user actions, uploading and uploading the ontology. It also supports direct manipulation with the service tables by means of SQL.

8. Conclusions and perspectives
Practice of implementation and numerous experiments show fruitfulness of the described approach to the data integration and search in structurally marked texts, which are seen here as multiply connected graphs, implicitly reflecting the semantics of the subject domain.

Defining a query, an expert user presumes some meaningful links between key words used in the definition. Thus, formulation of the query as well as structure of stored information implicitly reflect semantics of the subject domain. This combination helps the search engine to find complete and sufficiently relevant set of answers. That explains effectiveness of quite simple search algorithm implemented in described approach.

At the same time, the considered search mechanism has several obvious directions for further development. Say, important results could give analyses of real mechanisms of data normalization. The known normal forms, including the generalized normal form proposed by R. Fagin in 1981, do not change the set of the database attributes, but just redistribute them between tables. However, in the real databases, quite often a key with a small number of values is transformed to a group of attribute names or tables. Thus, the set of attribute values and the set of attribute names can be transformed into each other. De facto, this practice is reflected in some constructions of storage mechanisms of industrial databases — for example, in segmentation of large tables by keys, supported in Oracle, PostgreSQL and a number of other systems. To some extent, this practice has been also legitimized in repositories with the capability of integration of data from various sources, where attributes can be stored in a table by introducing an additional attribute "Attribute name". Thus, the practice of transformation of the attribute name into the value of some super-type exists, but so far
it hasn't received sufficient theoretical coverage. In our opinion, detailed study of this phenomenon can give a valid contribution to the theory of relational databases.

In the context of the main topic under consideration, further formalization of this restructuring phenomenon can provide a key to automating the construction of queries and data replication between transaction systems with different database structures, as well as with repositories, multidimensional OLAP systems and other storage formats, including those based on XML. Creation of search engines, based on such new theoretical principles, is especially important for Big Data, for which heterogeneity of information representation is inherent.

A In the context of the main topic under consideration, further development of such restructuring mechanisms can provide a key to automating the construction of queries and data replication between different databases, as well as between repositories, multidimensional OLAP systems and other storage formats, including those based on XML. Creation of search engines, based on such new theoretical principles, is especially important for Big Data, for which heterogeneity of information representation is inherent.

In the context of the main topic under consideration, further development of such restructuring mechanisms can provide a key to automating the construction of queries and data replication between different databases, as well as between repositories, multidimensional OLAP systems and other storage formats, including those based on XML. Creation of search engines, based on such new theoretical principles, is especially important for Big Data, for which heterogeneity of information representation is inherent.

An obvious possible obstacle to the described approach is put by its combinatorial complexity. However, continuously increasing data processing power can to great extent overcome this barrier.

Acknowledgments
This work was funded by the subsidy allocated to Kazan Federal University for the state assignment in the sphere of scientific activities, grant agreement no. 1.2368.2017.

References
[1] Hopkins B and Evelson B 2012 Expand your digital horizon with Big Data Available from: http://www.asterdata.com/newsletter-images/30-04-2012/resources/forrester_expand_your_digital_horiz.pdf
[2] NOSQL Databases Available from: http://nosql-database.org/
[3] Apache Hadoop, MapReduce Available from: https://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html
[4] Apache Hadoop Available from: http://hadoop.apache.org/
[5] SAP HANA Available from: https://www.sap.com/products/hana.html
[6] Kogalovskij M 2010 Methods of data integration in information systems (Moscow: Russian Academy of Science Institute of Market Problems) Available from: http://www.ipr-ras.ru/articles/kogalov10-05.pdf
[7] W3C, OWL Web Ontology Language Available from: https://www.w3.org/TR/2004/REC-owl-features-20040210/
[8] Birialtsev E, Gusenkov A and Kosinov Y 2007 Representation of relational database structure in the ontology formalism Proc. of Kazan School on Computer and Cognitive Linguistics TEL-2006 (Kazan: Otechestvo Publishing House) p 32–37
[9] Birialtsev E and Gusenkov A 2007 Integration of databases based on ontologies Scholarship Notes of the Kazan University. Physics and mathematics series 149 p 13–25
[10] Gusenkov A, Birialtsev E and Zhibrik O 2015 Intellectual search in structured information arrays (LAP LAMBERT Academic Publishing)
[11] Garcia-Molina H, Ullman J and Widom J 2003 Database Systems: The Complete Book (Williams Publishing House)
[12] Booch G, Rumbaugh J and Jacobson I 2000 *The Complete UML Training Course* (Moscow: DMK Press)

[13] Birialtsev E, Gusenkov A and Galimov M 2005 Features of the lexico-semantic structure of the relational databases artifacts naming *Proc. of Kazan School on Computer and Cognitive Linguistics TEL-2005* p 4–12

[14] Zhuchkov A, Arnautov S and Tverdohlebov N 2003 New technologies for conceptual networks created within framework of the project “New generation vaccines and diagnostic systems of the future” *Electronic Libraries* 6 Available from: [http://www.elbib.ru/index.phtml?page=elbib/rus/journal/2003/part6/ZATGS](http://www.elbib.ru/index.phtml?page=elbib/rus/journal/2003/part6/ZATGS)

[15] Gavrilova T and Horoshevskij V 2001 *Knowledge base of intelligent systems* (St. Petersburg: Piter Publishing House)

[16] Anderson J 2003 *Discrete Mathematics with Combinatorics* (Moscow: Williams Publishing House)

[17] Birialtsev E, Gusenkov A and Khairullina A 2006 Presentation of the POSC Epicenter data model in the OWL ontology language *Proc. of Kazan School on Computer and Cognitive Linguistics TEL-2006* p 38–49

[18] Birialtsev E and Gusenkov A 2007 The construction of subject domain ontology based on the logical database model *Proc. of Knowledge-Ontology-Theory (ZONT-07)* Conf. (Novosibirsk: Sobolev Mathematical Institute Press 1) p 176–183

[19] *Energistics, Epicentre v3.0* Available from: [http://www.energistics.org/energistics-standards-directory/epicentre-archive](http://www.energistics.org/energistics-standards-directory/epicentre-archive)

[20] Fellbaum C (ed.) 1998 *WordNet: An Electronic Lexical Database* (Cambridge: MIT Press)

[21] Birialtsev E and Gusenkov A 2007 Relational database ontologies. Linguistic aspect *Proc. Of Dialog’2007* Conf. P 50–53

[22] Narinyani A S 2001 Centaur named Theon: Thesaurus + Ontology *Proc. of the international workshop on computational linguistics and its applications DIALOG 1* p 184–188

[23] Birialtsev E, Gusenkov A and Mironov S 2009 One approach to implementing unregulated access to relational databases *Proc. of Kazan School on Computer and Cognitive Linguistics TEL-2008* p 10–23

[24] Chaudhuri S and Dayal U 1997 An Overview of Data Warehousing and OLAP Technology *ACM SIGMOD Record* **26**(1) p 65–74