

10. Calle Rosingana, G. (2012). *Perspectiva lingüística y cognitiva del estilo de Carlos Ruiz Zafon en La sombra del viento* [Linguistic and cognitive perspective of Carlos Ruiz Zafon's style in La sombra del viento] // URL: <https://www.tdx.cat/handle/10803/81112> (accessed: 12.12.2018) (In Spanish)

11. Ortiz, R. *Recorrido literario por Carlos Ruiz Zafon. Analisis critico de la Sombra del viento* [Literary tour by Carlos Ruiz Zafon. Critical analysis of the Shadow of the wind]. // URL: <https://ru.calameo.com/read/003622025d138ec0b1f0c> (accessed: 10.12.2018) (In Spanish)

12. Zafón, C.R. (2008). *El priveden Juego Del Ángel* [The priveden Game of the Angel]. Editorial Planeta, S. A. Diagonal, 662–664–08034. (In Spanish)

#### Авторы публикации

**Палутина Ольга Геннадьевна** – к.филол.н.,  
доцент кафедры европейских языков и  
культур Высшей школы иностранных языков  
и перевода Института международных  
отношений Казанского федерального  
университета, г. Казань, Россия.  
E-mail: [Olga.Palutina@ksu.ru](mailto:Olga.Palutina@ksu.ru)

#### Authors of the publication

**Palutina Olga Gennadyevna** – Associate  
Professor, PhD, Department of European  
Languages and Cultures, Higher School of  
Foreign Languages and Translation, Institute of  
International Relations, Kazan Federal  
University, Kazan, Russia.  
E-mail: [Olga.Palutina@ksu.ru](mailto:Olga.Palutina@ksu.ru)

Поступила в редакцию 10.11.2018.  
Принята к публикации 01.12.2018.

УДК 81

## МОДЕЛИРОВАНИЕ КОМПЬЮТЕРНОГО ТЕХНИЧЕСКОГО ЯЗЫКА С ПОМОЩЬЮ ГРАФИКОВ СВОЙСТВ

**Л. Р. Сакаева, Клегер Деспайне Э.**

[ecleger@uci.cu](mailto:ecleger@uci.cu)  
[dekleger@stud.kpfu.ru](mailto:dekleger@stud.kpfu.ru)

**Казанский (Приволжский) федеральный университет, г. Казань, Россия**  
**Университет информационных наук, г. Гавана, Куба**

**Аннотация.** Переработка естественного языка стала одной из основных проблем в информационном обмене. Быстрое развитие компьютеров позволило реализовать

множество идей для решения проблем, которые невозможно было даже представить себе автоматически, когда появились первые компьютеры. Интеллектуальная обработка естественного языка основана на науке, называемой вычислительной лингвистикой. Вычислительная лингвистика тесно связана с прикладной лингвистикой и лингвистикой в целом. В настоящее время исследователей в основном интересует формальное описание языка, имеющего отношение к автоматической обработке речи, а не чисто алгоритмические вопросы. По этой причине в данной статье мы представляем пример того, как моделировать компьютерный технический язык с помощью графиков свойств, а также описываем некоторые особенности графиков свойств с помощью базы данных Neo4J. Изложены преимущества использования графиков свойств для моделирования компьютерного технического языка. Наконец, мы выбрали пять свойств языка для представления в модели: Фонология, морфология, синтаксис, семантика и прагматика.

**Ключевые слова:** лингвистика, моделирование, компьютерный язык, графики свойств.

**Для цитирования:** Сакаева Л.Р., Клегер Деспайне Э. Моделирование компьютерного технического языка с помощью графиков свойств // Казанский лингвистический журнал. 2019, том 1, № 4 (4). С. 41–51.

## MODELING COMPUTER TECHNICAL LANGUAGE BY MEANS OF PROPERTY GRAPHS

*L.R. Sakaeva, Cleger Despaigne E.*

*ecleger@uci.cu  
dekleger@stud.kpfu.ru*

*Kazan Federal University, Kazan, Russia  
University of Informatics Sciences, Havana, Cuba*

**Abstract.** The processing of natural language has become one of the main problems in information exchange. The rapid development of computers has made possible the implementation of many ideas to solve the problems that one could not even imagine being solved automatically, when the first computers appeared. Intelligent natural language processing is based on the science called computational linguistics. Computational linguistics is closely connected with applied linguistics and linguistics in general. Currently, the researchers are mainly interested in the formal description of language relevant to automatic language processing, rather than in purely algorithmic issues. For this reason, in this paper, we present an example about how to model computer technical language by means property graphs, and also describes some features of property graphs using Neo4J database. We state the advantages of property graphs use for modeling computer technical language. Finally, we selected five language properties to represent in the model: Phonology, morphology, syntax, semantics, and pragmatics.

**Keywords:** Linguistics, modeling, computer language, property graphs.

**For citation:** Sakaeva L.R. Cleger Despaigne E., Modeling computer technical language by means of property graphs // Kazan linguistic journal. 2019. Vol. 1, No. 4 (4). Pp. 41–51.

The structure of our knowledge is so closely entwined with the structure of our language that it may seem foolhardy to try to disentangle them. Yet it is necessary to do so in order to make some assessment of the consequences of using language, especially the language of literate prose, as the predominant means of instruction in the schools. It is clear that knowledge may be acquired through a variety of means: private experience, observation of a model, or explicit instruction [1, p.328].

Saussure's *Course in General Linguistics*, published posthumously in 1916, stressed examining language as a static system of interconnected units. He is thus known as a father of modern linguistics for bringing about the shift from diachronic (historical) to synchronic (non-historical) analysis, as well as for introducing several basic dimensions of semiotic analysis that are still important today.

This means that language structure can't be conceived atomistically, that is: its elements (the signs) can be separated one from each other's. By this reason the structuralists defend the holistic perspective, that is: the idea that properties from a system can't be determined or explained from its isolated components. Thus, language structure is based both: on the differential relationship between terms and the fact that these terms cannot be understood without considering their interconnection [2, p.141].

Thus, the processing of natural language has become one of the main problems in information exchange. The rapid development of computers has made possible the implementation of many ideas to solve the problems that one could not even imagine being solved automatically, when the first computers appeared.

Intelligent natural language processing is based on the science called computational linguistics. Computational linguistics is closely connected with applied linguistics and linguistics in general. Therefore, we shall first outline shortly linguistics as a science belonging to the humanities.

Computational linguistics might be considered as a synonym of automatic processing of natural language, since the main task of computational linguistics is just the construction of computer programs to process words and texts in natural language.

But today “more linguistic than computational,” for the following reason: The researchers are mainly interested in the formal description of language relevant to automatic language processing, rather than in purely algorithmic issues. The algorithms, the corresponding programs, and the programming technologies can vary, while the basic linguistic principles and methods of their description are much more stable [3, p.25].

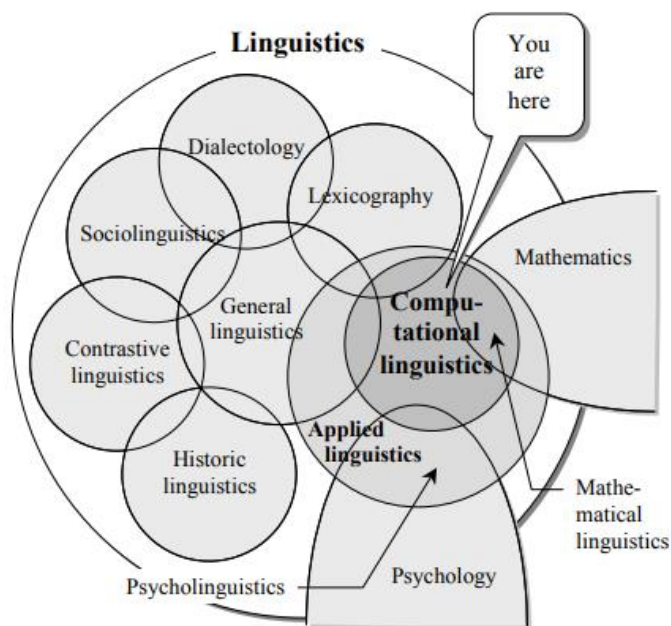


Figure 1. Structure of linguistic science [3, p.17]

In this paper we present an example modeling computer technical language by means property graphs, and also describes some features of property graphs using Neo4J database. In the last decade, property graph databases such as Neo4j, JanusGraph and Sparksee have become more widespread in industry and academia. They have been used in multiple domains, such as *master data* and *knowledge management*, *recommendation engines*, fraud detection, IT operations and network

management, authorization and access control, bioinformatics, *social networks*, software system analysis, and in investigative journalism. Using graph databases to manage graph-structured data confers many benefits such as explicit support for modeling graph data, native indexing and storage for fast graph traversal operations, built-in support for graph algorithms (e.g., Page Rank, subgraph matching and so on), and the provision of graph languages, allowing users to express complex pattern-matching operations [4, p.1434].

▪ **The advantages of property graphs use for modeling computer technical language**

A graph structure means that we will be using vertices and edges (or nodes and relationships, as we prefer to call these elements) to store data in a persistent manner. As a consequence, the graph structure enables us to [5, p.34]:

- Represent data in a much more natural way, without some of the distortions of the relational data model.
- Apply various types of graph algorithms on these structures.

The property graph model is optimized for:

- Directed graphs: The links between nodes (also known as the relationships) have a direction.
- Multirelational graphs: There can be multiple relationships between two nodes that are the same. These relationships, as we will see later, will be clearly distinct and of a different type. This feature is important for representing the relationships between words.
- Storing key-value pairs as the properties of the nodes and relationships.
- Walk on its nodes and relationships and hop from one node to the next by following the explicit pointers that connect the nodes. This capability—sometimes also referred to as index free adjacency, which essentially means that you can find adjacent/neighbors nodes without having to do an index lookup.

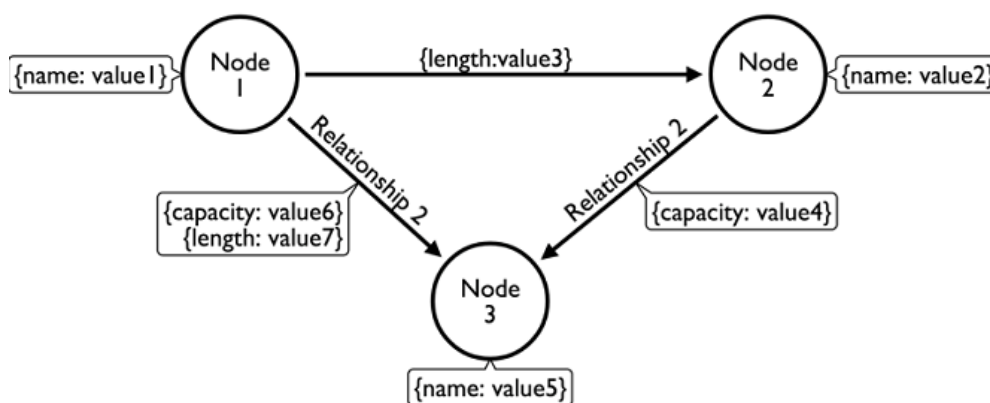


Figure 2. An example of a simple property graph [5, p.35]

Some Neo4J architecture advantages [6, p.83]:

- The node store is a fixed size record store where each record is 9 bytes in length. Fixed record sizes enable fast lookups for nodes within the store file – if we have a node with id 100 then we know its record begins 900 bytes into the file and so the database can directly compute the record location at cost  $O(1)$  without performing a search at cost  $O(\log n)$ .

- The relationship store contains fixed-size (in this case 33 bytes) records and each record contains the id of the nodes at the start and end of each relationship, a pointer to the relationship type (stored in the relationship type store) and pointers for the relationship chains for the start and end node (as a doubly linked list) since a relationship logically belongs to both nodes and therefore should appear in the list of both nodes' relationships.

Was performed by the author in [7, p.15] an experiment to establish the relationships between used model for data management and representation and its computational cost. Was analyzed also if the amount of data influenced the response times. It was evidence that using graph-oriented database reduces temporary complexity in scenarios of large volumes of data.

### Language Properties for modeling

As far as general linguistics is concerned, its most important parts are the following [3, p.17]:

- Phonology deals with sounds composing speech, with all their similarities and differences permitting to form and distinguish words.
- Morphology deals with the inner structure of individual words and the laws concerning the formation of new words from pieces - morphs.
- Syntax considers structures of sentences and the ways individual words are connected within them.
- Semantics and pragmatics are closely related. Semantics deals with the meaning of individual words and entire texts, and pragmatics studies the motivations of people to produce specific sentences or texts in a specific situation.

We selected five language properties to represent in the model, there exist others properties, but we are interested in those mentioned above.

### Proposed model by means of property graphs.

We already have the corresponding technology for modeling, and we define which language properties will be represented in our property graph. In the following figure we may see the structure our model.

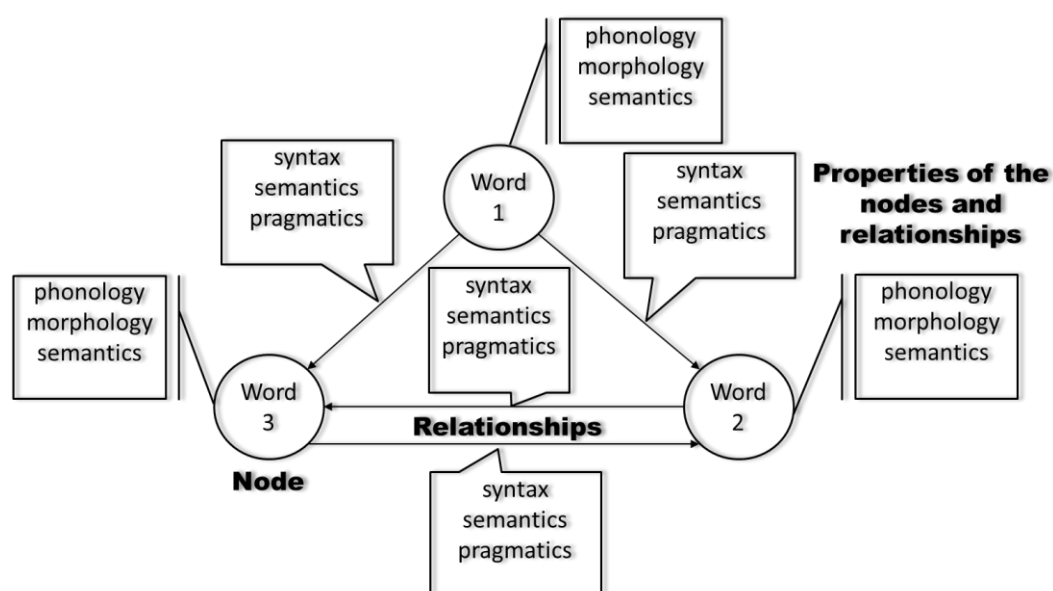


Figure 3. Model using property graphs

Each word exists independently and is represented in the graph by means of nodes. The relationships between words can be simple or complex (when you

extend more than one type of relationship between words). To each "word" node is added the properties: phonology, morphology and Semantics. In the same way each relation represents the properties: syntax, semantics and pragmatics. Relationships also have a label whose name represents the relationship type between nodes "words". Table 1 describes attributes and relationships of representing above model.

From the previous model and the description of its components, an example is presented with the term "client-server model". This term is composed of three words: model, client and server. The proposed graph-oriented model describes the existing relations between the words that constitute the term and the five properties of the language selected by the authors of this work. Node 01 "Model", node 02 "Client" and node 03 "Server" are presented. In addition, the relationship Type and the Join, about the latter is important to know that it aims to indicate that both words must be combined in the conformation of the term and its structure includes an intermediate dash.

Table 1

#### NODES, RELATIONSHIPS AND ATTRIBUTES OF PROPOSED GRAPH-BASED MODEL

NODE	ATTRIBUTE	DESCRIPTION
<b>word</b>	word_Id	Numeric word identifier
	phonology	audio file with sounds for distinguishing words.
	morphology	Character string with the inner structure of individual words and morphs.
	semantics	Character string with the word meaning.
<b>relationship</b>	relationship_Id	Numeric relationship identifier
	syntax	Character string with relationship structure and the way each word is connected.
	semantics	Character string with the relationship meaning
	pragmatics	Character string with the motivations to create relationships between words.



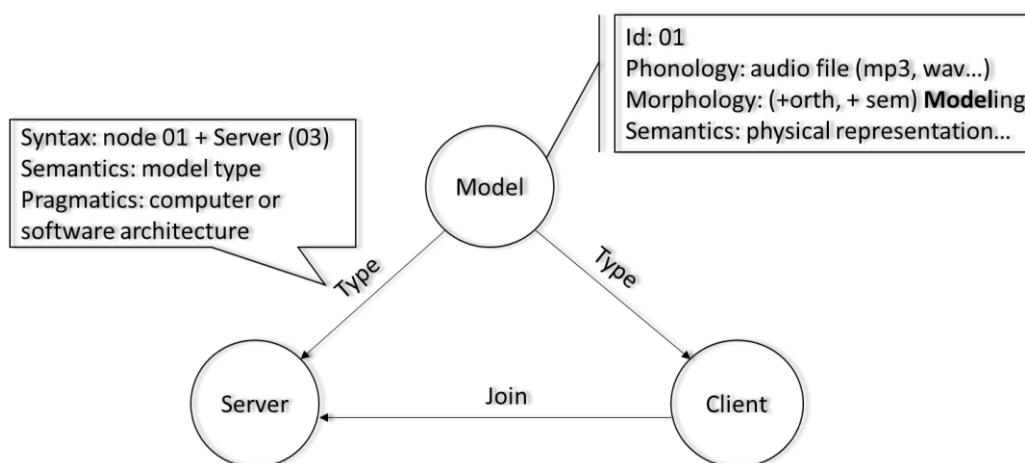


Figure 4. Example of the model using property graphs

If we observe the morphological property, we shall that the noun model can become in verb and share both a similar orthography and a related semantics (**Modeling**). From the semantic property, we can know the meaning or meanings of the word. The syntax property located in the relations between nodes, will indicate us the form in which the words are related to both ends of the relation (node 01 + node 03). The semantic property of the relation expresses the meaning of the label that names the relation between both nodes (model type). The pragmatic property gives us the contexts in which the term can be used (computer or software architecture).

To conclude, we'd like to underline that this approach for representing computers technical language by means of property graphs, provides a didactic tool for acquiring knowledge in the linguistic field.

In term of information technologies, this work becomes another way for data management optimization in systems related to computational linguistics.

### *Литература*

1. Anderson R., Spiro R., Montague W. Schooling and the Acquisition of Knowledge: Chapter 3. The Languages of Instruction: The Literate Bias of Schooling. London: Routledge, 1977. 443 p.

2. *Bolshakov I.A., Gelbukh A.* Computational linguistics: Models, Resources, Applications. Mexico: Ipn-Unam-Fce, 2004. 186 p.

3. *Cleger E.* et al. Graph Oriented Model on Neo4j. A free alternative to reduce temporary complexity // *Iberoamerican Journal of Project Management*. 2017. Vol. 8. No. 1. Pp. 1–17.

4. *Francis N., Green A., Guagliardo P, Libkin L., Lindaaker T.* Cypher: An Evolving Query Language for Property Graphs // *SIGMOD'18 Proceedings of the 2018 International Conference on Management of Data*. Jun 2018, Houston, United States. ACM Press, 2018. P.1433.

5. *Robinson I., Webber J., Eifrem E.* Graph Databases. Sebastopol: O'Reilly Media Inc., 2013. 205 p.

6. *Saussure F.* Course in General Linguistics. Losada: Open Court House, 1971.

7. *Van Bruggen R.* Learning Neo4j. Birmingham. Mumbai: Packt Publishing Ltd, 2014. 201 p.

### ***References***

1. Anderson, R., Spiro, R., Montague, W. (1977). *Schooling and the Acquisition of Knowledge*: Chapter 3 The Languages of Instruction: The Literate Bias of Schooling. London: Routledge, 443 p. (In English)

2. Bolshakov, I.A., Gelbukh, A., (2004). *Computational linguistics: Models, Resources, Applications*. Mexico: Ipn-Unam-Fce, 186 p. (In English)

3. Cleger, E. et al. (2017). *Graph Oriented Model on Neo4j. A free alternative to reduce temporary complexity* [Graph Oriented Model on Neo4j. A free alternative to reduce temporary complexity] // *Iberoamerican Journal of Project Management*. Vol. 8. No. Pp. 1-17. (In Spanish)

4. Francis, N., Green, A., Guagliardo, P, Libkin, L., and Lindaaker, T. (2018). *Cypher: An Evolving Query Language for Property Graphs* // *SIGMOD'18 Proceedings of the International Conference on Management of Data*. Jun 2018, Houston, United States. ACM Press, pp. 1433–1448. (In English)

5. Robinson, I., Webber, J., Eifrem, E. (2013). *Graph Databases*. Sebastopol: O'Reilly Media Inc., 205 p. (In English)
6. Saussure, F., Harris, R. (1998). *Course in General Linguistics*. Losada: Open Court House, 1971. (In English)
7. Van Bruggen, R. (2014). *Learning Neo4j*. Birmingham. Mumbai: Packt Publishing Ltd, 201 p. (In English)

#### Авторы публикации

**Сакаева Лилия Радиковна** – доктор филологических наук, профессор, зав. кафедрой иностранных языков для физико-математического направления и информационных технологий Высшей школы иностранных языков и перевода Института международных отношений Казанского федерального университета, г. Казань, ул. Межлаука, д. 3.  
Email: liliyasakaeva@rambler.ru

**Клегер Деспайне Элиобер** – Магистр в области прикладной информатики, доцент кафедры Программной инженерии Университета информационных наук, Гавана, Куба,  
E-mail: ecleger@uci.cu

#### Authors of the publication

**Liliya Radikovna Sakaeva** – Doctor in Philological Sciences, Professor, Head of the Department of Foreign Languages for the Physical and Mathematical Direction and Information Technologies of the Higher School of Foreign Languages and Translation of the Institute of International Relations of Kazan Federal University, 3, Mezhlauka Street, Kazan.  
Email: liliyasakaeva@rambler.ru

**Eliober Cleger Despaigne** – Master's Degree in Applied Informatics, Assistant Professor, Department of Software Engineering, University of Informatics Sciences, Havana, Cuba,  
E-mail: ecleger@uci.cu

Поступила в редакцию 10.11.2018.  
Принята к публикации 01.12.2018.