Integration of computer algebra tools into OSTIS applications

Valery B. Taranchuk
Department of Computer Applications and Systems
Belarusian State University
Minsk, Republic of Belarus
taranchuk@bsu.by

***Abstract*—From the standpoint of the need for convergence and unification of intelligent computer systems of a new generation, the issues of technology development and modernization, integration of the OSTIS metasystem with the Wolfram Mathematica computer algebra system are discussed. The current results and plans for the semantic representation of abstract mathematics on the Wolfram Language platform are noted as benchmarks. The software solutions implemented in the Wolfram Knowledgebase are marked and illustrated with examples.**

***Index Terms*—Semantic analysis, Wolfram Mathematica, Wolfram Knowledgebase, Entity**

I. Introduction

Following the assessment given in [1] of the current state of work in the field of Artificial Intelligence (AI), we can note the active local development of various areas (non-classical logic, formal ontologies, artificial neural networks, machine learning, soft computing, multi-agent systems, etc.), but there is no comprehensive increase in the level of intelligence of modern intelligent computer systems. What is especially relevant at the moment? Convergence and integration of all areas of Artificial Intelligence and the corresponding construction of a general formal theory of intelligent computer systems (ICS) are required.

It is important to transform the modern variety of tools (frameworks) for the development of various components of ICS into a single technology for integrated design and support of the full life cycle of these systems, which guarantees the compatibility of all components being developed, as well as the compatibility of ICS themselves as independent entities interacting with each other within complex automation systems of complex types of collective human activity. Convergence and unification of intelligent computer systems of the new generation and their components is necessary. At the same time, convergent solutions mainly mean optimized complexes containing everything necessary to solve AI problems, organized or configured for the effective use of information resources, to simplify implementation processes; in particular, it should be possible to solve certain tasks with optimization requirements and achieve maximum performance, and in all implementations – optimized for ease of use. The key reasons of methodological problems of the current state of Artificial Intelligence, as well as a number of required actions to solve them are indicated in [1]. Supporting these concepts, we note that such problems are solved when developing, improving, systematically updating the content and expanding the capabilities of computer algebra systems. The following examples illustrate several methodological and technical solutions for convergence and integration of various types of knowledge implemented in the computer algebra system Wolfram Mathematica (WM), Wolfram Language (WL).

II. Wolfram Mathematica. The semantic representation of pure mathematics. The current state, plans

The current state. Based on more than thirty years of research, development and use throughout the world, Mathematica and the Wolfram Language are aimed at a long-term perspective and are particularly successful in computational mathematics. About 6000 characters embedded in the Wolfram Language allow you to represent and manipulate a huge variety of computational objects in the system – from special functions to graphics and geometric areas. In addition, the Wolfram Knowledgebase [2] and the associated entity structure [3] allow you to explain/interpret/formalize hundreds of specific "things (facts/situations/objects)". For example: people, cities, food, structures, planets, ... are represented by objects that can be manipulated, they can be cheated.

Wolfram Mathematica. Immediate plans. Despite a rapidly and ever-increasing number of domains known to WL, many knowledge domains still await computational representation. In his blog "Computational Knowledge and the Future of Pure Mathematics" Stephen Wolfram presented a grand vision for the representation of abstract mathematics, known variously as the Computable Archive of Mathematics or Mathematics Heritage Project (MHP). The eventual goal of this project is no less than to render all of the approximately 100 million pages of peer-reviewed research mathematics published over the last several centuries into a computer-readable form.

Wolfram Mathematica. The semantic representation of pure mathematics. In the blog [4], leading Wolfram specialists present their vision of the future semantic representation of abstract mathematics using two examples: abstract mathematical concepts of functional, topological spaces; concepts and theorems of general topology. It seems that such concepts and approaches should be used in solving methodological problems of the current state of Artificial Intelligence.

III. The Wolfram Knowledgebase. Examples

Powering Wolfram|Alpha and WL, the ever-growing Wolfram Knowledgebase (WKB) is by far the world's largest and broadest repository of computable knowledge. WKB, covering thousands of fields, contains carefully selected expert knowledge directly derived from primary sources ([4]). It includes not only trillions of data elements, but also immense number of algorithms encapsulating methods and models from almost every field. The main subject fields of WKB are illustrated in Fig. 1.

Figure 1. The main subject fields of WKB.

The Wolfram Knowledgebase relies on three decades of computable knowledge acquisition. All data in the Wolfram database can immediately be used for computation in WL. Every millisecond of every day, WKB is updated with the latest data.

Let's note a few examples. With a trove of statistics for hundreds of thousands of educational institutions around the world, Wolfram|Alpha can compute answers to intricate questions about education You can request which academic degrees students of prestigious universities receive. You can also compute the average salary for teachers in your local school district, learn more about student scores, and compare student-teacher ratios among countries and much more. Fig. 2 illustrates the response to the request for the number of students in the Republic of Belarus.

The comparison for the universities of BSU and BSUIR is illustrated in Fig. 3.

Figure 2. How many high school students are there in Belarus.

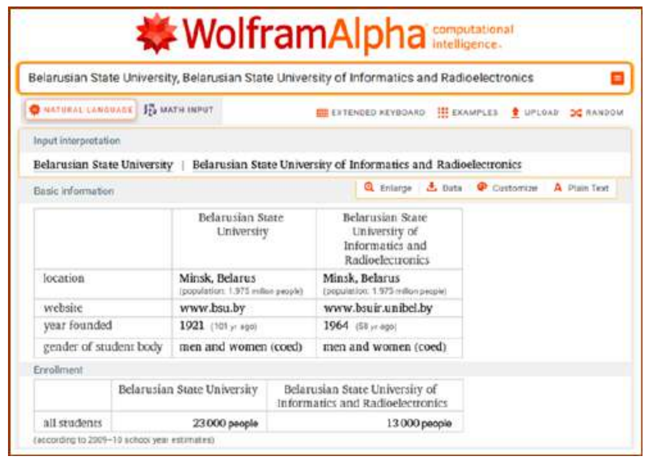


Figure 3. How many high school students are there in Belarus.

IV. The Wolfram Knowledgebase. Representation and Access to It

Access to WKB is deeply integrated into WL. Free-form linguistics makes it easy to identify many millions of entities and many thousands of properties and automatically generate precise WL representations suitable for extensive further computation. The Wolfram Language also supports custom entity stores that allow the same computations as the built-in knowledgebase, and can be associated with external relational databases. Note the main groups of WM functions for working with WKB: Entity & EntityClass & EntityValue, Transformations & Computations on Entity Classes, Standard Properties, Specific Domains, Setting Up Custom Entity Stores, Wolfram Data Repository, Wolfram Data Drop, Setting Up Custom Entity Stores, External Knowledgebases, External Database Connectivity, Web Content, Textual Question Answering, System Configuration. There are more than three subgroups in each of the listed groups. For example, the Textual Question Answering group includes:

- *FindTextualAnswer* attempt to find answers to questions from text;
- *SemanticInterpretation* convert free-form linguistics to Wolfram Language for; *SemanticInterpretation*["string"] attempts to give the best semantic interpretation of the specified free-form string as a Wolfram Language expression;
- *SemanticImport* import data, converting entities etc. to Wolfram Language form,
- *Interpreter* interpret input of various types (e.g. "City", "Date", etc.); Interpreter attempt to interpret strings of a wide variety of types; *Interpreter*[form] represents an interpreter

object that can be applied to an input to try to interpret it as an object of the specified form.

V. Semantic Analysis

Humans interact with each other through speech and text, and this is called Natural language. Computers understand the natural language of humans through Natural Language Processing (NLP). NLP is a process of manipulating the speech of text by humans through Artificial Intelligence so that computers can understand them. Human language has many meanings beyond the literal meaning of the words. There are many words that have different meanings, or any sentence can have different tones like emotional or sarcastic. It is very hard for computers to interpret the meaning of those sentences.

NLP. Main applications, tools implemented in the Wolfram Mathematica system: Speech Recognition, Voice Assistants and Chatbots, Auto Correct and Auto prediction, Email Filtering, Sentiment Analysis, Divertissements to Targeted Audience, Translation, Social Media Analytics, Recruitment, Text Summarisation.

Several representative examples with explanations of the functions of the WL groups Structural Text Manipulation, Text Analysis, Natural Language Processing are mentioned below. In a sense, these categories are conditional, there are a lot of functions and capabilities implemented by them. For example, the Structural Text Manipulation subgroup may include the following: TextCases – extract symbolically specified elements (*TextCases*[text,form] gives a list of all cases of text identified as being of type form that appear in text); TextSentences – extract a list of sentences (*TextSentences*["string"] gives a list of the runs of characters identified as sentences in string); TextWords – extract a list of words (*TextWords*["string"] gives a list of the runs of characters identified as words in string); SequenceAlignment – find matching sequences in text; *TextStructure*["text"] generates a nested collection of TextElement objects representing the grammatical structure of natural language text [5].

An example and the result of executing the TextStructure function to the text "Open Semantic Technologies for Intelligent Systems" with the option "ConstituentTree" is shown in Fig. 4

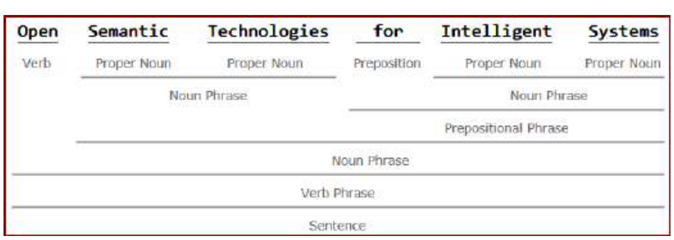


Figure 4. The result of executing the TextStructure function with the option "ConstituentTree".

Variants for visualizing the components of the analyzed phrase with the settings "ConstituentGraphs", "DependencyGraphs" are shown in Fig. 5.

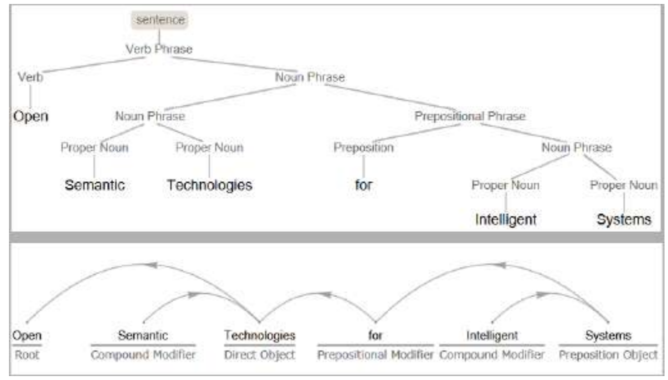


Figure 5. Results of executing the TextStructure function with the settings "ConstituentGraphs", "DependencyGraphs".

NLP. Examples of using the FindTextualAnswer function. Answer questions in natural language from the text [6].

FindTextualAnswer[text,"question",n] gives a list of up to n answers that appear most probable. *FindTextualAnswer*[text,"question",n,prop] gives the specified property for each answer.

In the two examples below, the processing object is text *International scientific and technical conference proceedings "Open Semantic Technologies for Intelligent Systems (OSTIS)". Established: 2011. Scientific areas of the conference: 05.13.11, 05.13.15, 05.13.17*. In the query variant with the option "Date of establishment of the conference?" the answer is

"2011"

In the request variant with the options "Date of the conference establishment?", "Scientific directions of the conference?" the answer is a list

"2011", "05.13.11, 05.13.15, 05.13.17"

In the following example, the processing object is text *International scientific and technical conference proceedings Open Semantic Technologies for Intelligent Systems (OSTIS). Established: 2011. Scientific areas of the conference: Theory of Informatics, Software for Computers, Computer Complexes and Networks, Computing Machines and Complexes and Computer Networks*.

In the query variant with the option "Scientific areas of the conference?" the answer is –

"Theory of Informatics, Software for Computers, Computer Complexes and Networks, Computing Machines and Complexes and Computer Networks".

Next example illustrates the search in the text and the fixation of three signs. The object of processing is text *International scientific and technical conference proceedings Open Semantic Technologies for Intelligent Systems (OSTIS). Established: 2011. Program Committee: Kuznetsov Oleg Co-Chair, Dr. of Techn. Sciences, Professor, Academician of the Russian*

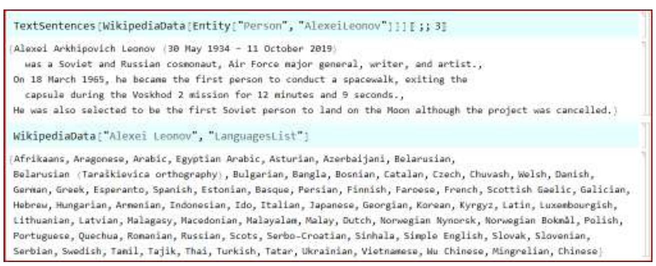
Academy of Natural Sciences, Moscow, Russia; Golenkov Vladimir Co-Chair, Dr. of Techn. Sciences, Professor, Minsk, Belarus; ... Arefiev Igor Dr. of Techn. Sciences, Professor, Szczecin, Poland; ... Globa Larisa Dr. of Techn. Sciences, Professor, Kyiv, Ukraine ... .

In the query variant with the options "City", "Country", "Date", the response is –

<|"City" -> "Moscow", "Minsk", "Szczecin", "Kyiv", "Country" -> "Russian", "Russia", "Belarus", "Poland", "Ukraine", "Date" -> "2011"|>

VI. EXAMPLES OF EXTRACTING KNOWLEDGE, ENTITIES, OR TOPICS FROM WIKIPEDIA ARTICLES

WikipediaData utilizes MediaWiki's API to retrieve article and category contents and metadata from Wikipedia [7]. An article may be specified as a string or a Wolfram Language entity. The extraction of articles associated with language entities is provided by the WM TextSentences function, in particular, you can work with Wikipedia resources. *TextSentences*["string"] gives a list of the runs of characters identified as sentences in string [8]. *WikipediaData*[article] gives the plain text of the specified Wikipedia article. *Entity*["type",name] represents an entity of the specified type, identified by name. *WikipediaData*[article,property,options] gives the value of the specified property, modified by optional parameters, for the given Wikipedia article. Below are the results of executing the *TextSentence* function, with the parameters WikipediaData, Entity, "Person", "AlexeiLeonov" and displaying a list of language versions of Wikipedia (*LanguagesList*) containing the corresponding article (in Fig. 6)


```
TextSentences[WikipediaData[Entity["Person", "AlexeiLeonov"]]][[;; 3]]
{Alexei Arkhipovich Leonov (30 May 1934 – 11 October 2019)
  was a Soviet and Russian cosmonaut, Air Force major general, writer, and artist.,
On 18 March 1965, he became the first person to conduct a spacewalk, exiting the
  capsule during the Voskhod 2 mission for 12 minutes and 9 seconds.,
He was also selected to be the first Soviet person to land on the Moon although the project was cancelled.}

WikipediaData["Alexei Leonov", "LanguagesList"]
{Afrikaans, Aragonese, Arabic, Egyptian Arabic, Asturian, Azerbaijani, Belarusian,
Belarusian (Taraškievica orthography), Bulgarian, Bangla, Bosnian, Catalan, Czech, Chuvash, Welsh, Danish,
German, Greek, Esperanto, Spanish, Estonian, Basque, Persian, Finnish, Faroese, French, Scottish Gaelic, Galician,
Hebrew, Hungarian, Armenian, Indonesian, Ido, Italian, Japanese, Georgian, Korean, Kyrgyz, Latin, Luxembourgish,
Lithuanian, Latvian, Malagasy, Macedonian, Malayalam, Malay, Dutch, Norwegian Nynorsk, Norwegian Bokmål, Polish,
Portuguese, Quechua, Romanian, Russian, Scots, Serbo-Croatian, Sinhala, Simple English, Slovak, Slovenian,
Serbian, Swedish, Tamil, Tajik, Thai, Turkish, Tatar, Ukrainian, Vietnamese, Wu Chinese, Mingrelian, Chinese}
```

Figure 6. TextSentences. WikipediaData. Entity. AlexeiLeonov. LanguagesList.

Fig. 7 illustrates the system's response to the execution of the WikipediaData function with the parameters "Voskhod 2", "ImageList":

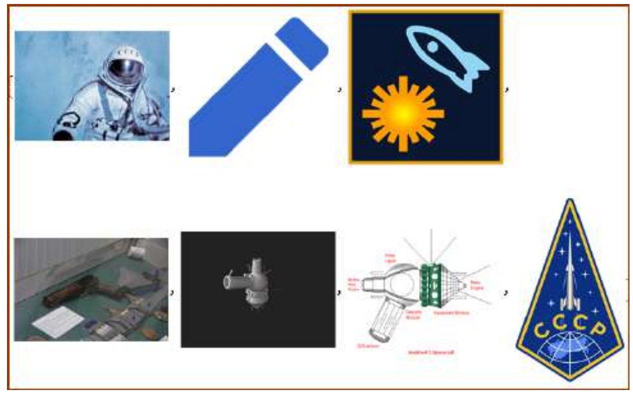


Figure 7. TextSentences. WikipediaData. Entity. Voskhod 2. ImageList.

List of rules representing links between categories. *WikipediaData*["Category"->category,property,options] gives the value of the specified property, modified by optional parameters, for the given Wikipedia category. "MaxLevelItems" – number of links to follow at each level. "MaxLevel" – number of levels to search outward from the specified page.

Fig. 8 shows the result of the function *WikipediaData*["Category"->"Artificial intelligence", "CategoryLinks", "MaxLevelItems"->5, "MaxLevel"->3] execution in the form of a graph:

You can output all categories separately, for example, for Applications of artificial intelligence, the system will output: "Category:Agent-based software", "Category:Applied data mining", "Category:Applied machine learning", "Category:Automated planning and scheduling", "Category:Computer vision software".

Examples of **extracting in the knowledge cloud, entities or topics, lists of rules representing relationships between categories** are given in the description of the WordCloud function. (*WordCloud*[s_1,s_2,...] generates a word cloud graphic in which the s_i are sized according to their multiplicity in the list.) The illustration in Fig. 9 is obtained in Wolfram Mathematica, the functions Delete stop words and Text Words are used. (*DeleteStopwords*[list] deletes stopwords from a list of words; *TextWords*["string"] gives a list of the runs of characters identified as words in string).

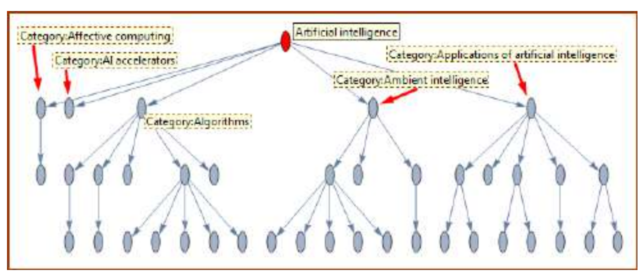


Figure 8. Graph for "Artificial intelligence".

VII. INTELLECTUALIZATION OF USER INTERFACES. EXAMPLES OF IMPLEMENTATION IN WOLFRAM MATHEMATICA

Currently, the user interfaces of many computer systems (including intelligent computer systems) in most implementations are not semantically friendly. For users, interaction with computer systems is often a "bottleneck" that has a significant impact on the efficiency of automation of human activities. The basis of the modern organization of user interaction with a computer system is the paradigm of a trained, competent user

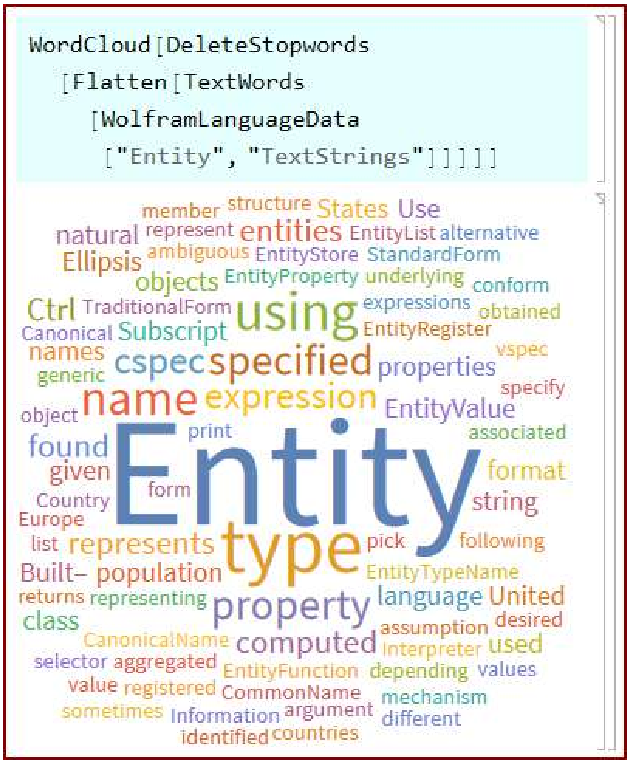


Figure 9. WordCloud for "Entity".

who knows the capabilities of the tool he uses, is responsible for the correctness of interaction with him. At the present stage of the development of Artificial intelligence, in order to increase the efficiency of interaction, it is necessary to move from the paradigm of competent management of the tool used to the paradigm of equal cooperation, partnership interaction of an intelligent computer system with the user. Semantic friendliness of the user interface should consist in adaptability to the features and qualifications of the user, eliminating problems for the user in the process of dialogue with the computer system. It is fundamental to switch from a friendly user interface to an intelligent predictive interface, in which the system, when working with a computer, not only speeds up the input of queries, simplifies the dialogue, clarifies the correctness of commands and actions, but also offers, after the output of the next result, a line of options for the following calculations, actions. Moreover, for each user, such a line of hints (prompts) is formed taking into account the accumulated statistics of individual preliminary requests and the content of the system, and in each particular session it includes links to several possible formalization approaches, several different scenarios of work on examples of similar tasks, virtual textbooks. According to the accumulated knowledge in the navigation block of the line of hints, the user can go to illustrations of other recommended processing methods by means of computer mathematics (with reference to the subject area being performed), interpretation algorithms.

Explanations, several typical illustrations, fragments of the windows of the user interface of the Mathematica system, in which the intelligent predictive interface is implemented, are shown in the following figures 10, 12.

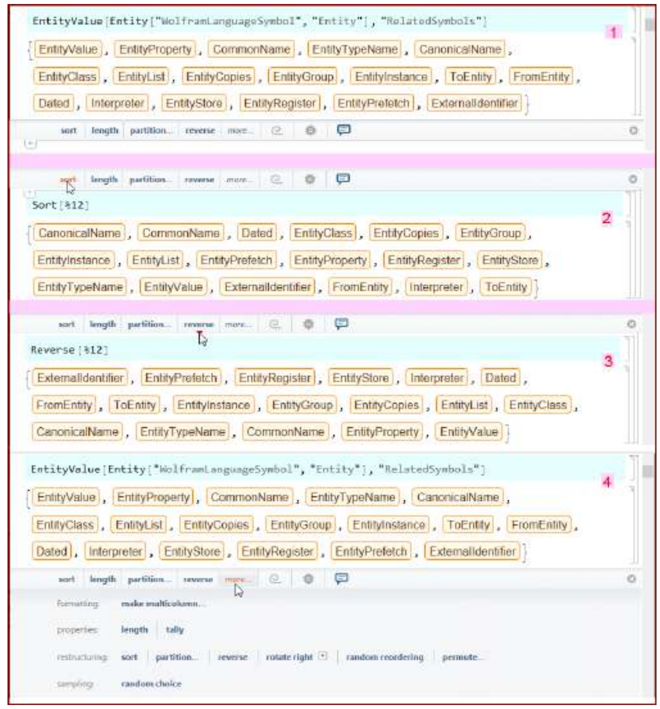


Figure 10. EntityValue. Entity. RelatedSymbols..

Each illustration includes several fragments of copies of the screens. Examples are given for the above request to display a cloud of documentation words for Entity, and for related WM functions associated to Entity [5], [6]. As a result of executing the section with the *EntityValue*[Entity["WolframLanguageSymbol", "Entity"], "RelatedSymbols"] command, the user receives an answer and a line of suggestions for further actions. Similar tooltips are displayed after each section is completed, and in all cases their content is determined by semantic connections. Fragment 2 Fig. 10 – the functions found and shown are sorted alphabetically, the contents of fragment 3 – the functions are listed in reverse order, fragment 4 contains an expanded menu of possible further actions.

All functions of the Wolfram Mathematica system are documented in detail; clicking on the function name in any section of the system provides access to a notebook-article – description. The examples given in the description can be executed in the notebook-article itself, you can copy and transfer to any other notebook, there are no restrictions on replacing data and settings.

Figure 11 shows a fragment of one (randomly selected) of the functions mentioned in the list – EntityProperty.

The illustration in Fig. 11, sections of the article-descriptions are typical; the structure is the same for everyone. It is important that the examples are grouped by difficulty levels, but there is no need to perform them sequentially.

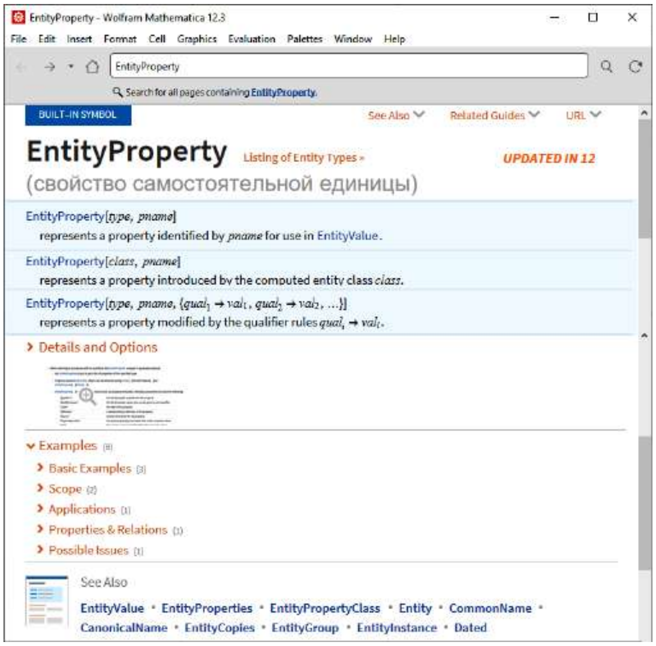


Figure 11. EntityProperty. Article-description (notebook document) structure

Figure 12 shows the steps of work when the country (United States) is changed (Belarus) in the basic example, and then queries are selected following the prompts in the tooltips.

It should be noted that it is also possible to state certain successes in solving problems of underground hydrodynamics [9] , forest fires [10] on the formation and filling of knowledge bases with Wolfram Mathematica tools during processing, accumulation and interpretation of the results of computational experiments.

VIII. Conclusion

Several representative examples of working with knowledge bases by means of the Wolfram Mathematica system are discussed. Since the functions of the Mathematica core can be used in programs developed on other platforms, presented results can be interpreted as proposals for innovative improvement of existing tools, components of intelligent computer systems.

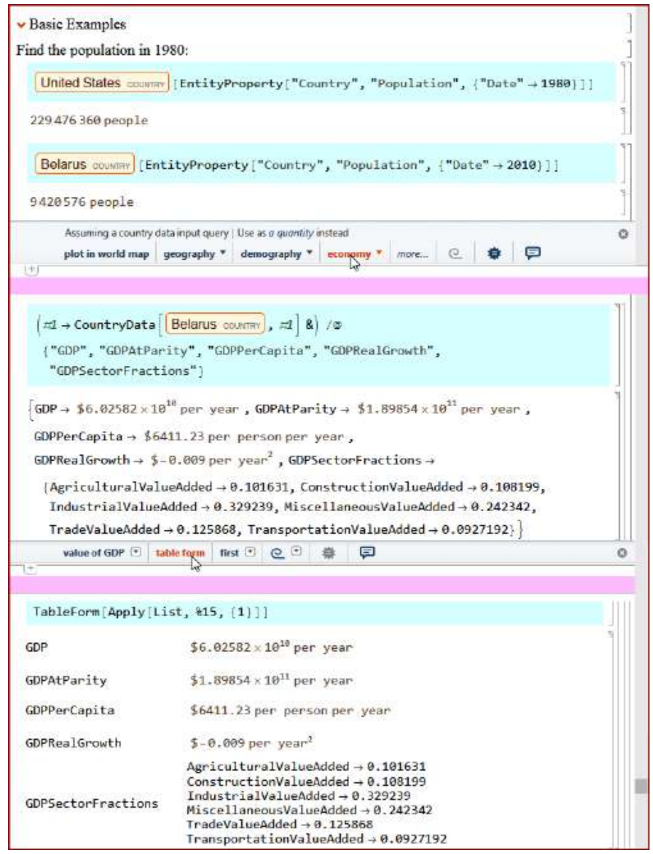


Figure 12. An illustration of the use of a ruler (palette) of suggestions for the following calculations.

References

[1] V. Golenkov, N. Guliakina, and D. Shunkevich, Otkrytaja tehnologija ontologicheskogo proektirovanija, proizvodstva i jekspluatacii semanticheski sovmestimyh gibridnyh intellektual'nyh komp'juternyh sistem [Open technology of ontological design, production and operation of semantically compatible hybrid intelligent computer systems], V. Golenkov, Ed. Minsk: Bestprint [Bestprint], 2021.

[2] Wolfram Knowledgebase. Making the knowledge of the world computable. Available at: https://www.wolfram.com/knowledgebase (accessed 2022, Oct).

[3] Knowledge Representation & Access. Wolfram Language & System Documentation Center. Available at: https://reference.wolfram.com/language/guide /KnowledgeRepresentationAndAccess.html (accessed 2022, Oct).

[4] The Semantic Representation of Pure Mathematics. Available at: https://blog.wolfram.com/2016/12/22/the-semantic-representation-of-pure-mathematics/ (accessed 2022, Oct).

[5] TextStructure. Available at: https://reference.wolfram.com/language/ref/TextStructure.html/ (accessed 2022, Oct).

[6] FindTextualAnswer. Available at: https://reference.wolfram.com/language/ref/FindTextualAnswer.html/ (accessed 2022, Oct).

[7] WikipediaData. Available at: https://reference.wolfram.com/language/ref/WikipediaData.html/ (accessed 2022, Oct).

[8] TextSentences. Available at: https://reference.wolfram.com/language/ref/TextSentences.html/ (accessed 2022, Oct).

[9] V. Taranchuk, Tools and examples of intelligent processing, visualization and interpretation of GEODATA, Modelling and Methods of Structural Analysis. IOP Conf. Series: Journal of Physics: Conf. Series Vol. 1425 (2020) 012160. – P. 9, doi:10.1088/1742-6596/1425/1/012160.

[10] D.V. Barovik, V.B Taranchuk. Tools for the analysis and visualisation of distributions and vector fields in surface forest fires modelling. Journal of the Belarusian State University. Mathematics and Informatics. – 2. – 2022. – P. 82-93.

Интеграция инструментов компьютерной алгебры в приложения OSTIS

Таранчук В. Б.

С позиций необходимости конвергенции и унификации интеллектуальных компьютерных систем нового поколения обсуждаются вопросы технологии разработки и модернизации, интеграции средств метасистемы OSTIS с системой компьютерной алгебры Wolfram Mathematica.

Received 15.11.2022