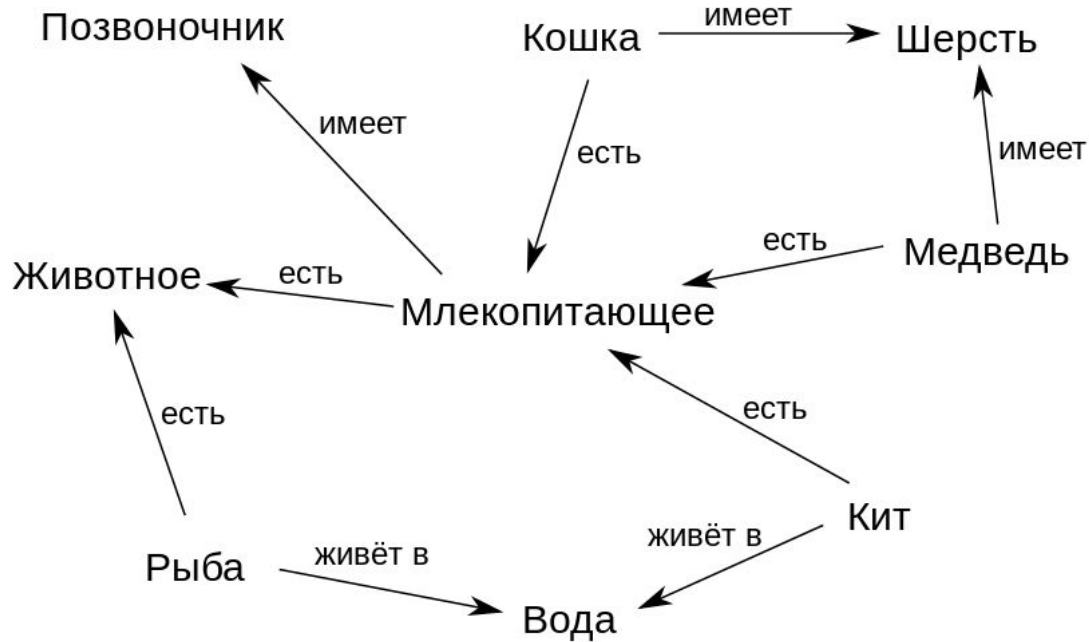


Prospects of using the ontology description language ONTOL V2 in the automation of the educational process

Pestryakov Danil
Peter the Great St. Petersburg Polytechnic University
17.07.2025

Представление онтологий. Семантическая сеть



Представление онтологий. RDF/RDFS/OWL

@prefix : <...some URL...>

@prefix rdf:

<http://www.w3.org/1999/02/rdf-schema#>

@prefix rdfs:

<http://www.w3.org/2000/01/22-rdf-syntax-ns#>

:Whale rdf:type :Mammal;

:livesIn :Water.

:Fish rdf:type :Animal;

:livesIn :Water.

Нотация RDF (Resource Description Framework).

Граф

:Mammal rdfs:subClassOf :Animal.

:Animal rdf:type rdfs:Class.

:Mammal rdf:type rdfs:Class.

:livesIn rdfs:range :Environment.

:Water rdf:type :Environment.

:Environment rdf:type rdfs:Class

Нотация RDFS. Описание иерархии классов и отношений

DisjointClasses(:Mammal :Fish)

Нотация OWL (Ontology Web Language).

Описание ограничений. К примеру, ни одно животное не может быть одновременно рыбой и млекопитающим.

Представление онтологий. UML vs RDF/RDFS/OWL

Преимущества UML

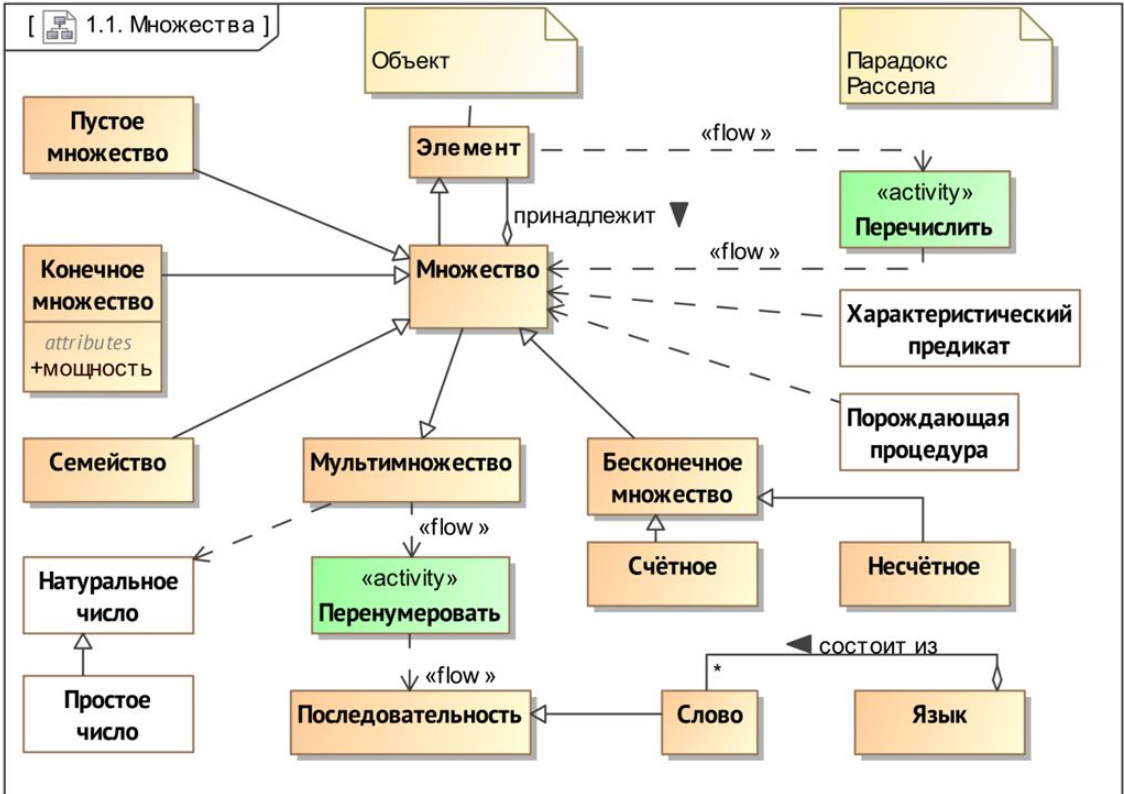
- ✓ Агрегация/композиция – встроенные отношения "часть–целое".
- ✓ Зависимости – гибкое обозначение связей без строгой формализации.
- ✓ Мета модель UML описана на UML.

[Novikov F. A., Molotkov I. I. Ontology of Discrete Mathematics in Education // Computer Tools in Education. 2021. No. 1. P. 5–16. (in Russian)]

Преимущества OWL (Ontology Web Language)

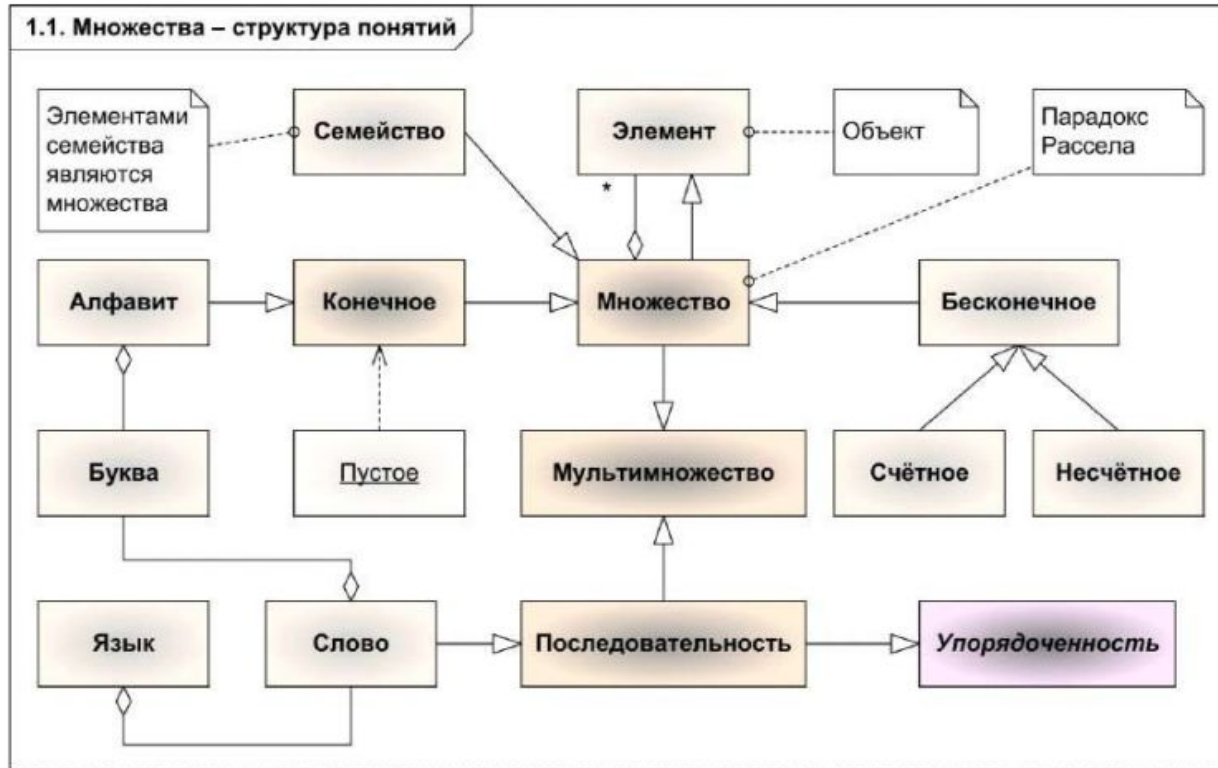
- ✓ Основан на дескрипционной логике.
- ✓ Точность – строгий вывод зависимостей.
- ✓ Стандартизация – совместимость с семантическими веб-технологиями.

Онтологии дискретной математики. Общая онтология множеств образца 2019 года



[Novikov F. A. Discrete Mathematics: Textbook for Universities. 2nd ed. Third Generation Standard. — St. Petersburg: Peter. 2013. 432 p.: illus. (in Russian)]

Онтологии дискретной математики. Понятийная грань



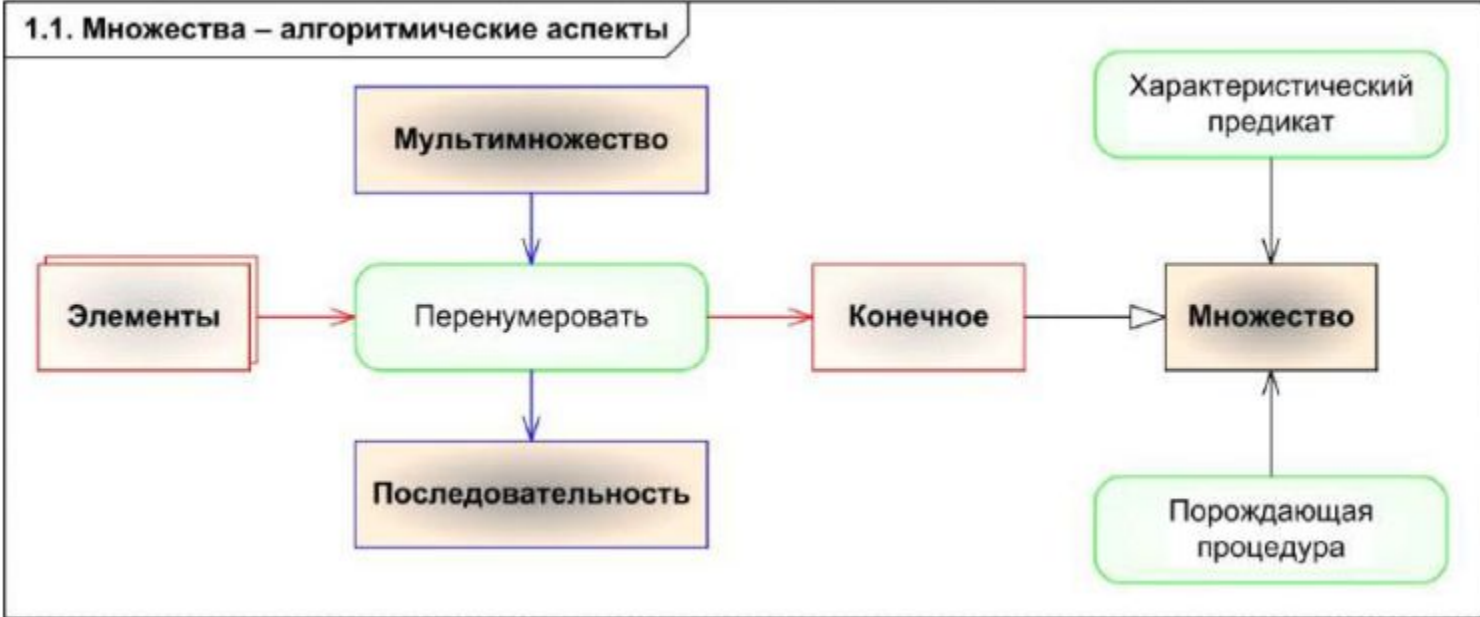
[Novikov F. A., Molotkov I. I. Ontology of Discrete Mathematics in Education // Computer Tools in Education. 2021. No. 1. P. 5–16. (in Russian)]

Онтологии дискретной математики. Грань зависимостей



[Novikov F. A., Molotkov I. I. Ontology of Discrete Mathematics in Education // Computer Tools in Education. 2021. No. 1. P. 5–16. (in Russian)]

Онтологии дискретной математики. Алгоритмическая грань

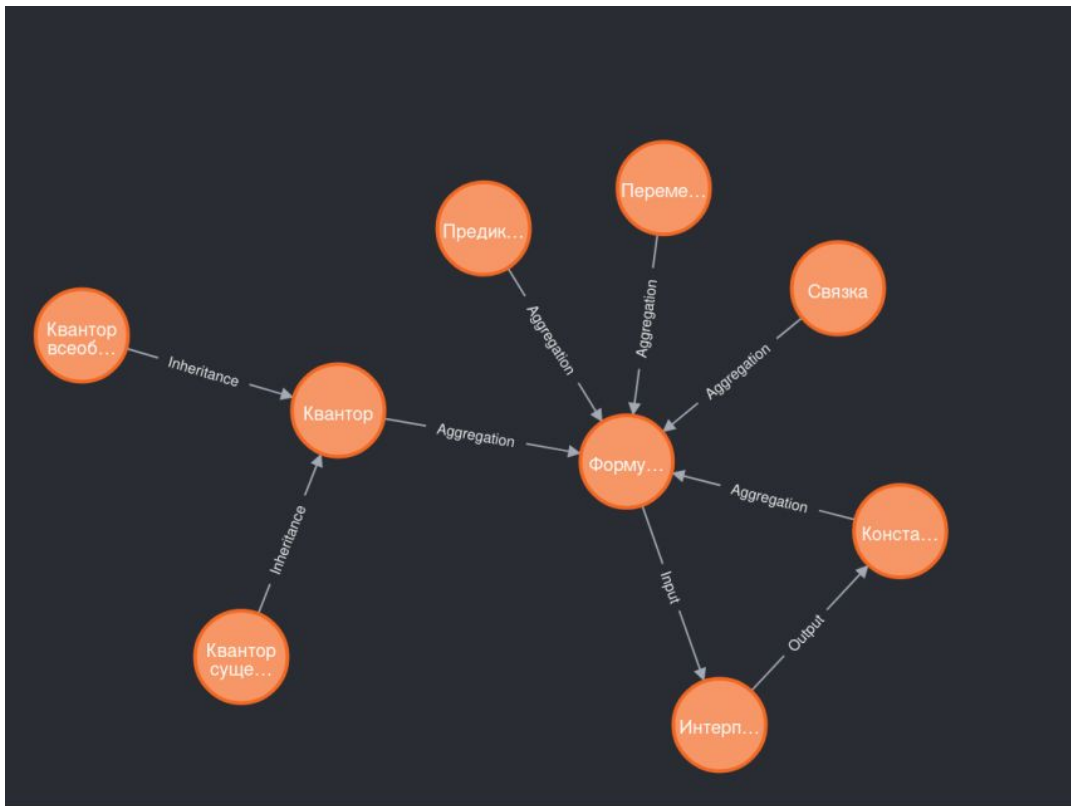


[Novikov F. A., Molotkov I. I. Ontology of Discrete Mathematics in Education // Computer Tools in Education. 2021. No. 1. P. 5–16. (in Russian)]

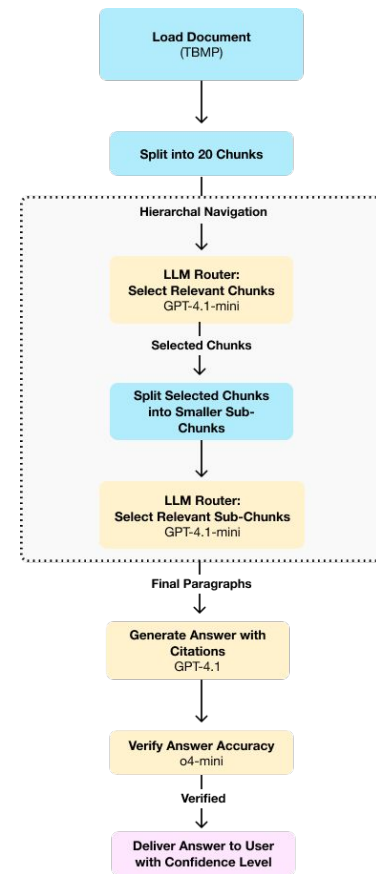
Перспективы развития. Создание текстового языка

1. Построение графов знаний
2. Инструмент с двусторонним действием для визуализации онтологий
3. Автоматизация процесса проведения промежуточной и итоговой аттестации

Графы знаний. LLM. Neo4j. Обзор



[Воротников А. А., ВКР магистра, СПбПУ, 2025]



[Practical Guide for Model Selection for Real-World Use Cases, OpenAI, 2025]

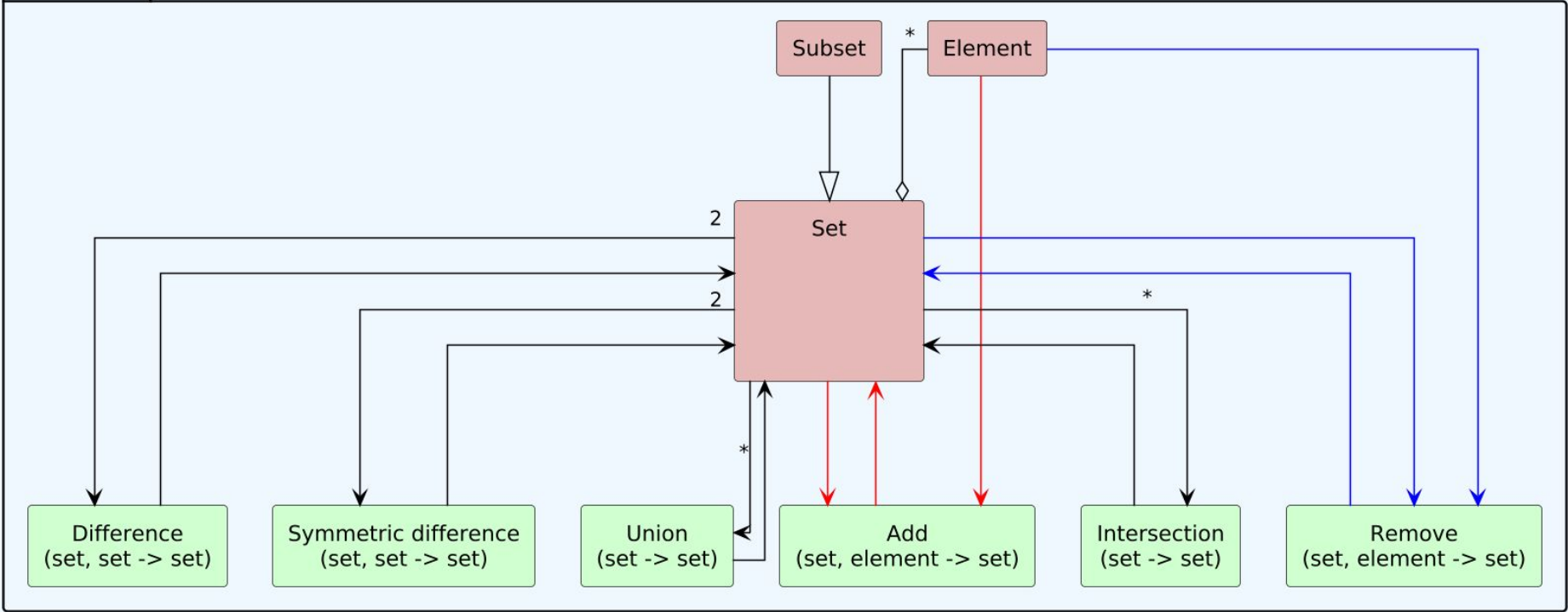
Язык описания онтологий ONTOL V1. Март 2025

```
1 version: '1.0'
2 title: 'Set theory'
3 author: ''
4 description: ''
5
6 types:
7 element: 'Element', '', { color: '#E6B8B7' }
8 set: '      Set      \n\n\n\n\n', '', { color: '#E6B8B7' }
9 subset: 'Subset', '', { color: '#E6B8B7' }
10
11 functions:
12 add: 'Add' (set: '', element: '') -> set: '', { color: '#D0FFD0', colorArrow: '#red' }
13 remove: 'Remove' (set: '', element: '') -> set: '', { color: '#D0FFD0', colorArrow: '#blue' }
14
15 union: 'Union' (set: '') -> set: '', { color: '#D0FFD0', inputTitle: '*' }
16 intersection: 'Intersection' (set: '') -> set: '', { color: '#D0FFD0', inputTitle: '*' }
17 difference: 'Difference' (set: '', set: '') -> set: '', { color: '#D0FFD0' }
18 symDiff: 'Symmetric difference' (set: '', set: '') -> set: '', { color: '#D0FFD0' }
19
20 hierarchy:
21 element aggregation set, { leftChar: '*' }
22 subset inheritance set
23
```

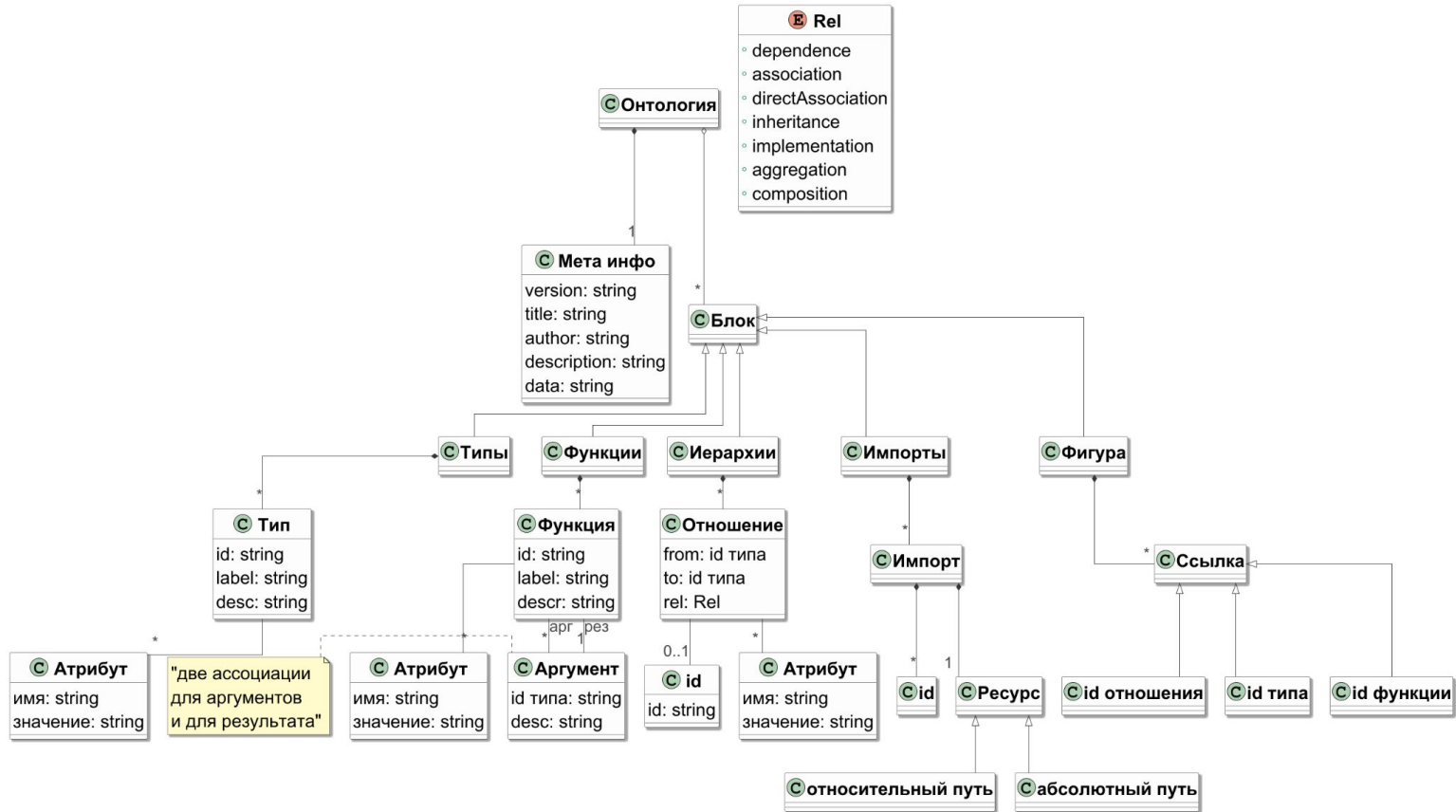
M. I. Aptukov, M. D. Markov,
F. A. Novikov, D. D.
Pestryakov, V. S. Skvortsov, I.
I. Khamidullin, “Ontology
description language ONTOL
V1,” Computer tools in
education, no. 1, pp. 91–107,
2025 (in Russian)

Язык описания онтологий ONTOL V1

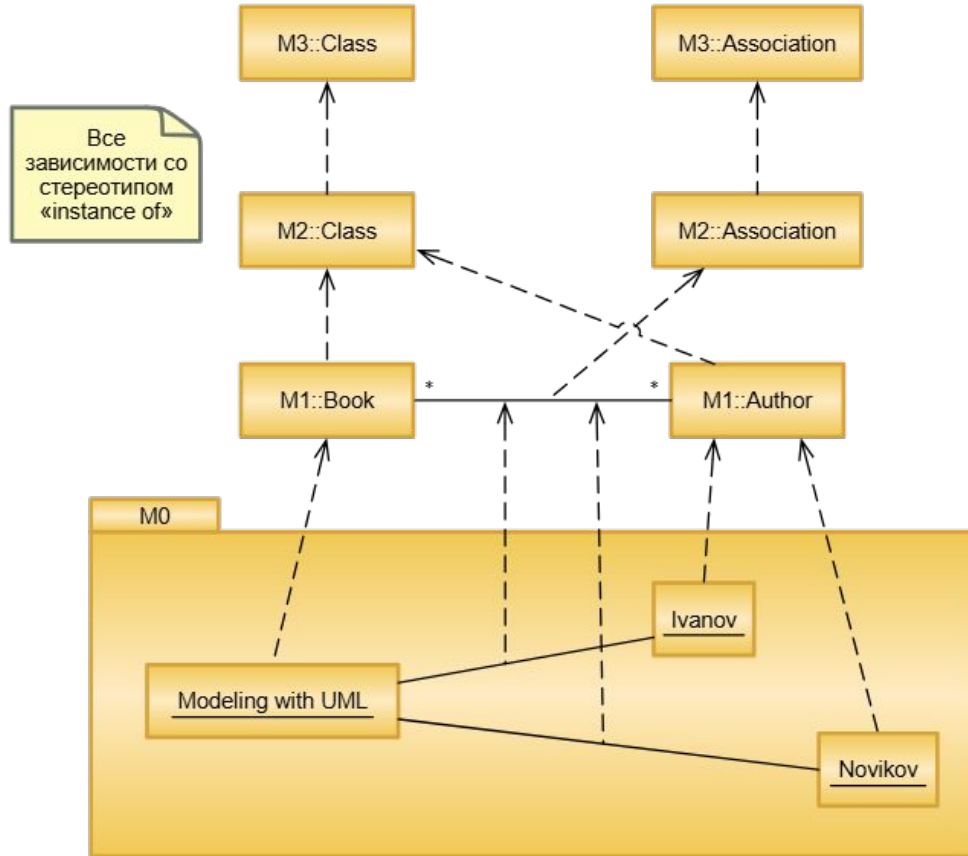
Set theory



Метамодель языка ONTOL V1



Метамоделирование



[Novikov F.A., Ivanov D.Yu. Modeling in UML. Theory, Practice, Video Course. — St. Petersburg, Professional Literature, Science and Technology, 2010]

[Terekhov, A. N. "Library introductions and separate translation of procedures in a translator from Algol 68." Abstracts of reports and messages for the All-Union conference. Part. Vol. 2. 1980. (in Russian)]

Описание онтологии СИИ на языке ONTOL V2. Июнь 2025


sai.ontolmy:
person:
 type: class.abstract
 realizations:
 Вейценбаум
 Уивер
object:
 type: class.abstract
 realizations:
 ELIZA
 Машинный перевод
dependence:
 type: relation.dependence ...
 realizations:
 творение
 основоположник

sai.sofy:
person:
 color: green
 coordinates:
 x: 0
 y: 0
 height: 10
 weight: 10
 Вейценбаум:
 coordinates:
 x: 10
 y: 10
end of sai.sofy

sai.ontol:
metamodels:
 sai.ontolmy
styles:
 sai.sofy
triplet_1:
 from: ELIZA
 to: Вейценбаум
 rel: творение
 stereotype: from_to
triplet_2:
 from: Машинный перевод
 to: Уивер
 rel: основоположник
 stereotype: to_from

Прекрасная ELIZA
Программа, понимающая естественный язык (1966)

Джозеф Вейценбаум (Joseph Weizenbaum),
1923-2008
известный деятель в области
искусственного интеллекта




«Существуют задачи, выполнение которых не
следует поручать вычислительным машинам,
независимо от того, можно ли добиться, чтобы
вычислительные машины их решали»

Ф.А. Новиков. Тема 1. Прикладные системы ИИ. 6 из 45

Машинный перевод

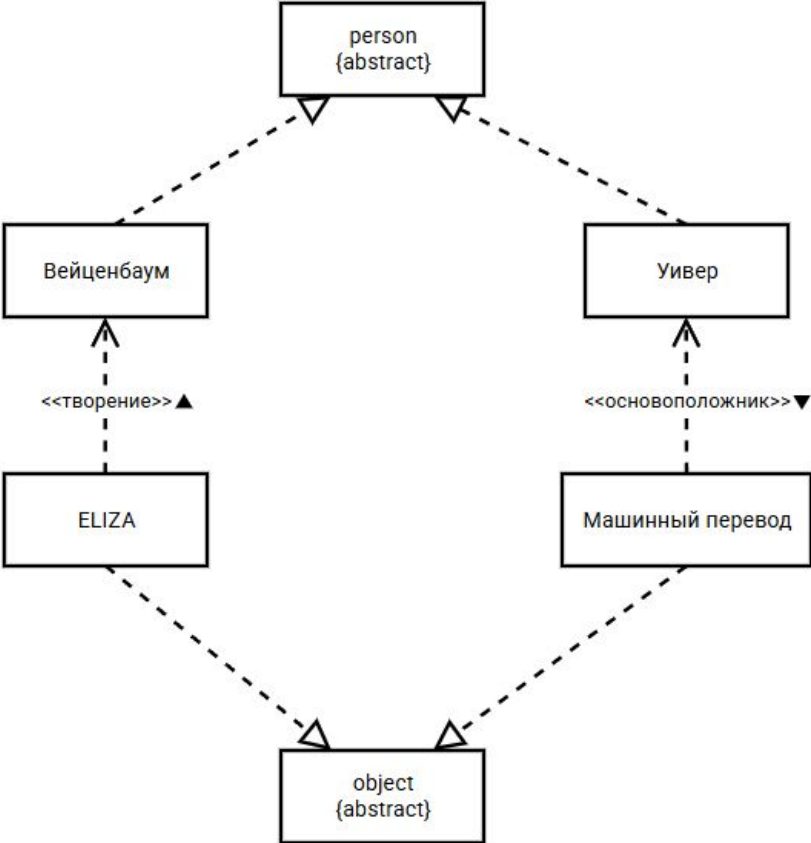
Уоррен Уивер (Warren Weaver),
1894-1978
изложил принципы современного
машинного перевода в 1947 г



«У меня перед глазами текст, написанный по-
русски, но я собираюсь сделать вид, что на самом
деле он написан по-английски и закодирован при
помощи довольно странных знаков.
Все что мне нужно — это взломать код!»

Ф.А. Новиков. Тема 1. Прикладные системы ИИ. 7 из 45

Диаграмма онтологии СИИ



Язык описания диаграмм d2

Достоинства:

- Свободно распространяемое программное обеспечение
- Инструмент визуализации диаграмм с двунаправленным действием
- Содержит модуль описания UML-диаграмм

Недостатки:

- Допускает ошибки при построении UML-диаграмм (пример: кратность обобщения)
- Облачная версия доступна в платном режиме

[D2 repository [Online]. Available: <https://github.com/terrastruct/d2> (accessed: 05.07.2025)]

Онтологии теорем. theorem.ontolmy

natInt

type: class.abstract

realizations:

натуральное число:

aliases:

NatInt

irdFrac:

type: class.abstract

realizations:

несократимая дробь:

aliases:

IrdFrac

sqrt:

type: class.abstract

realizations:

квадратный корень:

aliases:

Sqrt

axioms:

type: theorem.axiom

realizations:

аксиома

inputs:

type: theorem.input

realizations:

дано

outputs:

type: theorem.output

realizations:

доказать

theorems:

type: theorem

realizations:

теорема об иррациональности

[https://coq.vercel.app/examples/sqrt_2.html]

composition:

type: relation.composition

realizations:

необходимая часть:

pole: to_from

proofs:

type: theorem.proof

realizations:

эталонное доказательство

доказательство студента

Онтологии теорем. База кодов. theorem.ontol

triplet_1:

from: натуральное число

to: квадратный корень

rel: необходимая часть

code_1:

lang: python

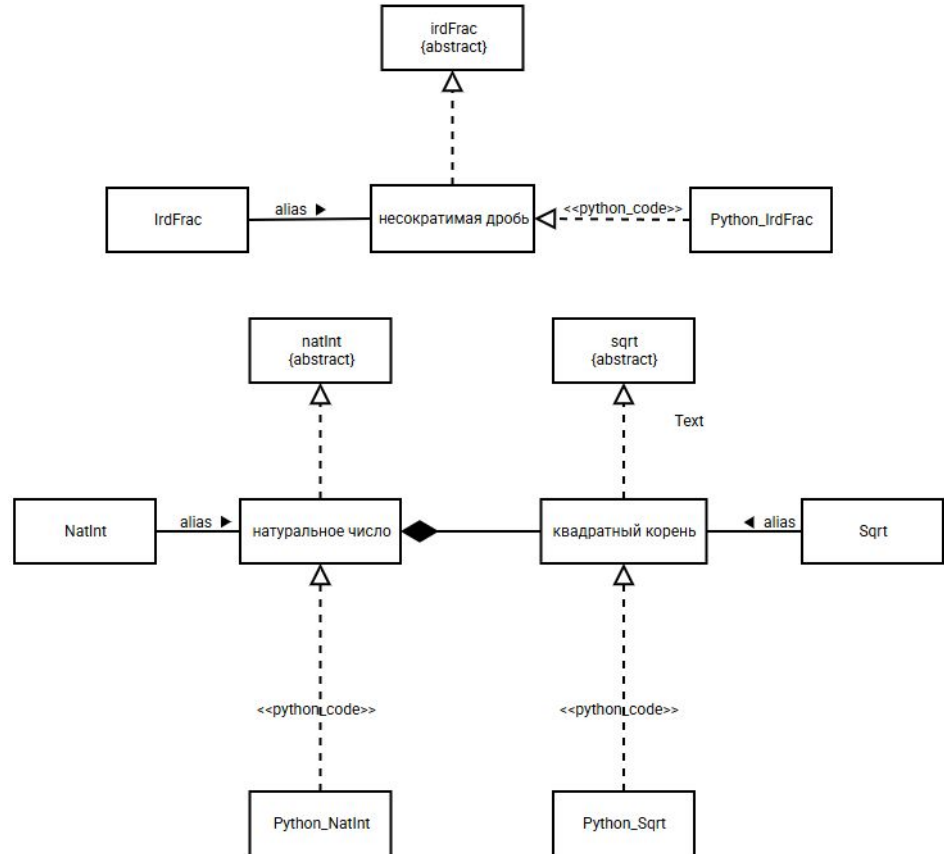
class: Sqrt

code: >

```
class Sqrt:
```

```
    def __init__(self, a: NatInt):
```

```
        self.a = current.a
```



Онтология теорем. theorem.ontol. Ассерты == аксиомы

```
theorem_1:
  theorem: теорема об иррациональности
  rel: дано
  notation: python
  expr: input = Sqrt(NatInt(2))

theorem_2:
  theorem: теорема об иррациональности
  rel: доказать
  notation: python
  expr: >
    rationals = IrdFrac()
    not any(input.a * q * q == p * p for p, q in rationals)

theorem_3:
  theorem: теорема об иррациональности
  rel: аксиома
  notation: python
  expr: m == n => NatInt(m).get_count_of_devider() == NatInt(n).get_count_of_devider()

theorem_4:
  theorem: теорема об иррациональности
  rel: аксиома
  notation: python
  expr: NatInt(q**2).get_count_of_devider() == 2 * NatInt(q).get_count_of_devider()

theorem_5:
  theorem: теорема об иррациональности
  rel: аксиома
  notation: python
  expr: NatInt(q).get_count_of_devider() + 1 == NatInt(2 * q).get_count_of_devider()
```

Онтология теорем. Пример доказательства

theorem_6:

theorem: теорема об иррациональности

rel: эталонное доказательство

notation: python

expr: >

1. $\text{any}(\text{input.a} * q * q == p * p \text{ for } p, q \text{ in rationals}) \Rightarrow p^{**2} == 2 * q^{**2}$

2. $\text{NatInt}(p^{**2}).\text{get_count_of_devider}() == 2 * \text{NatInt}(p).\text{get_count_of_devider}() \Rightarrow$
 $\text{NatInt}(p^{**2}).\text{get_count_of_devider}() \% 2 == 0$

3. $\text{NatInt}(q^{**2}).\text{get_count_of_devider}() == 2 * \text{NatInt}(q).\text{get_count_of_devider}() \Rightarrow$
 $\text{NatInt}(q^{**2}).\text{get_count_of_devider}() \% 2 == 0 \Rightarrow (\text{NatInt}(q^{**2}).\text{get_count_of_devider}() + 1) \% 2 == 1$

4. $\text{NatInt}(2 * q^{**2}).\text{get_count_of_devider}() == \text{NatInt}(q^{**2}).\text{get_count_of_devider}() + 1 \Rightarrow$
 $\text{NatInt}(2 * q^{**2}).\text{get_count_of_devider}() \% 2 == 1$

5. $p^{**2} == 2 * q^{**2} \Rightarrow$

$\text{NatInt}(p^{**2}).\text{get_count_of_devider}() == \text{NatInt}(2 * q^{**2}).\text{get_count_of_devider}() \Rightarrow$

$\text{NatInt}(2 * q^{**2}).\text{get_count_of_devider}() \% 2 == 1 \text{ and } \text{NatInt}(2 * q^{**2}).\text{get_count_of_devider}() \% 2 == 0 \Rightarrow$

False

Онтология теорем. Добавление доказательства в базу

theorem_7:

theorem: теорема об иррациональности

type: theorem.proof

rel: доказательство от Иванова Ивана Ивановича

tag: Иванов Иван Иванович

notation: python

expr: >

1. $\text{any}(\text{input.a} * q * q == p * p \text{ for } p, q \text{ in rationals}) \Rightarrow p^{**2} == 2 * q^{**2}$

2. $\text{NatInt}(p^{**2}).\text{get_count_of_devider}() == 2 * \text{NatInt}(p).\text{get_count_of_devider}() \Rightarrow$

$\text{NatInt}(p^{**2}).\text{get_count_of_devider}() \% 2 == 0$

3. $\text{NatInt}(q^{**2}).\text{get_count_of_devider}() == 2 * \text{NatInt}(q).\text{get_count_of_devider}() \Rightarrow$

$\text{NatInt}(q^{**2}).\text{get_count_of_devider}() \% 2 == 0 \Rightarrow (\text{NatInt}(q^{**2}).\text{get_count_of_devider}() + 1) \% 2 == 1$

4. $\text{NatInt}(2 * q^{**2}).\text{get_count_of_devider}() == \text{NatInt}(q^{**2}).\text{get_count_of_devider}() + 1 \Rightarrow$

$\text{NatInt}(2 * q^{**2}).\text{get_count_of_devider}() \% 2 == 1$

5. $p^{**2} == 2 * q^{**2} \Rightarrow$

$\text{NatInt}(p^{**2}).\text{get_count_of_devider}() == \text{NatInt}(2 * q^{**2}).\text{get_count_of_devider}() \Rightarrow$

$\text{NatInt}(2 * q^{**2}).\text{get_count_of_devider}() \% 2 == 1 \text{ and } \text{NatInt}(2 * q^{**2}).\text{get_count_of_devider}() \% 2 == 0 \Rightarrow$

False

Автоматическая проверка решений. ToulST. SAT

The screenshot displays the ToulST SAT solver interface. On the left, a sidebar contains navigation options: Connectors, Big ops (with $\bigwedge_{i \in a}$ and $\bigvee_{i \in a}$ buttons), Sets (with *atmost*, *atleast*, and *exact* buttons), and Other/Config. The main editor shows a problem statement in French with logical constraints. On the right, the logical translation is shown in CNF format, with tabs for Logical, CNF, DIMACS, and LaTeX. The CNF translation includes:

- $NATIONALITIES \leftarrow [englishman, spaniard, ukrainian, norwegian, japanese]$
- $ANIMALS \leftarrow [dog, oiswaterx, fox, horse, zebra]$
- $m \in HOUSES \rightarrow \bigvee_{c \in COLORS} Couleur_{m,c}$
- $m_1, c_1, c_2 \in HOUSES, COLORS \rightarrow \neg(Couleur_{m,c_1} \wedge Couleur_{m,c_2})$
- $m_1, m_2, c \in HOUSES, COLORS \rightarrow \neg(Couleur_{m_1,c} \wedge Couleur_{m_2,c})$
- $m \in HOUSES \rightarrow \bigvee_{g \in CIGARETTES} Fume_{m,g}$
- $m_1, g_1, g_2 \in HOUSES, CIGARETTES \rightarrow \neg(Fume_{m,g_1} \wedge Fume_{m,g_2})$
- $m_1, m_2, g \in HOUSES, CIGARETTES \rightarrow \neg(Fume_{m_1,g} \wedge Fume_{m_2,g})$
- $m \in HOUSES \rightarrow \bigvee_{b \in DRINKS} Boit_{m,b}$

[O. Gasquet, D. Longin, E. Lorini, F. Maris, P. R'egnier, and S. Soloviev, "ToulST, a Teacher-and Student-Friendly Language for Propositional Logic and Discrete Mathematics," Computer tools in education, no. 2, pp. 13–25, 2021]



Автоматическая проверка решений. TWF. Цепочки символьных преобразований. Июнь 2021

Write your solution instead of the dots and click the button 'Check solution'. Necessary formulas (cheat not for the learning process)

1 + i = $\sqrt{2} \cdot \left(\frac{1}{\sqrt{2}} + \frac{i}{\sqrt{2}} \right) = \sqrt{2} \left(\cos\left(\frac{\pi}{4}\right) + \sin\left(\frac{\pi}{4}\right)i \right) = \sqrt{2} \left(\frac{\exp(i \cdot \frac{\pi}{4}) + \exp(-i \cdot \frac{\pi}{4})}{2} + \sin\left(\frac{\pi}{4}\right)i \right) =$

$\sqrt{2} \left(\frac{\exp(i \cdot \frac{\pi}{4}) + \exp(-i \cdot \frac{\pi}{4})}{2} + \frac{\exp(i \cdot \frac{\pi}{4}) - \exp(-i \cdot \frac{\pi}{4})}{2i} \cdot i \right) = \sqrt{2} \left(\frac{\exp(i \cdot \frac{\pi}{4}) + \exp(-i \cdot \frac{\pi}{4})}{2} + \frac{\exp(i \cdot \frac{\pi}{4}) - \exp(-i \cdot \frac{\pi}{4})}{2} \right) =$

$\sqrt{2} \exp(i \cdot \frac{\pi}{4}) = \exp(\ln(\sqrt{2})) \exp(i \cdot \frac{\pi}{4}) = \exp(\ln(\sqrt{2}) + \frac{\pi i}{4})$

 *Solution is correct* 

[Novikov, Fedor & Katsman, Viktor & Mosin, Vladislav. (2020). Automated verification of expression transformation chains based on computational experiments. IOP Conference Series: Materials Science and Engineering. 734. 012132.]

Заключение

1. Спроектирован язык описания онтологий ONTOL V2, который применим для текстового описания:
 - a. Графов знаний
 - b. UML-диаграммы классов
 - c. Кодовых баз
 - d. Онтологии теорем
2. В язык ONTOL добавлен слой описания метамodelей, что позволяет:
 - a. Исследователям более точно описывать предметные области
 - b. Обучающимся читать онтологии на понятном им языке

Литература

1. Carta S. et al. Iterative zero-shot llm prompting for knowledge graph construction //arXiv preprint arXiv:2307.01128.– 2023.
2. Wang J. et al. Boosting language models reasoning with chain-of-knowledge prompting //arXiv preprint arXiv:2306.06427.– 2023.
3. Peng B. et al. Graph retrieval-augmented generation: A survey //arXiv preprint arXiv:2408.08921.– 2024.
4. Luo L. et al. Reasoning on graphs: Faithful and interpretable large language model reasoning //arXiv preprint arXiv:2310.01061.– 2023.
5. Novikov, F., Katsman, V., 2018. Gamification of problem solving process based on logical rules. In Informatics in Schools. Fundamentals of Computer Science and Software Engineering, Pozdniakov, S. N. & Dagien'e, V. (eds). Scopus, Springer International Publishing, 369–380.

Prospects of using the ontology description language ONTOL V2 in the automation of the educational process

Pestryakov Danil
Peter the Great St. Petersburg Polytechnic University
17.07.2025