

Министерство образования Республики Беларусь

Учреждение образования  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет Информационных технологий и управления  
Кафедра Интеллектуальных информационных технологий

**ОТЧЁТ**  
по ознакомительной практике

Выполнил:

З. А. Мацукевич

Студент группы  
421703

Проверил:

Н. В. Зотов

Минск 2025

## СОДЕРЖАНИЕ

Введение . . . . .	3
1 Формализация и сравнительный анализ языков представления информации в интеллектуальных системах: SKOS и OWL . . . . .	4
2 Формализация и сравнительный анализ инструментов для разработки интеллектуальных систем: Amazon Web Services (AWS) AI и Google Cloud AI . . . . .	9
Заключение . . . . .	14
Список использованных источников . . . . .	15

## **ВВЕДЕНИЕ**

### **Цель:**

Провести сравнительный анализ современных технологий разработки интеллектуальных систем, закрепить навыки формализации информации, а также приобрести и развить навыки исследования, анализа и сравнения технологических решений.

### **Задачи:**

1. Изучение языков представления знаний (SKOS и OWL) как инструментов формализации декларативных знаний, провести анализ их особенностей, выявить сходства и различия между ними.
2. Рассмотреть современные инструменты для работы с онтологиями и семантическими моделями (Amazon Web Services (AWS) AI / Neptune и Google Cloud AI), выявить их функциональные возможности и подходы к представлению знаний, сходства и отличительные особенности.
3. Формализовать полученные результаты в виде SCn-кода.
4. Оформить библиографические источники к полученным результатам по ГОСТ 7.1-2003.

# 1 ФОРМАЛИЗАЦИЯ И СРАВНИТЕЛЬНЫЙ АНАЛИЗ ЯЗЫКОВ ПРЕДСТАВЛЕНИЯ ИНФОРМАЦИИ В ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМАХ: SKOS И OWL

## SKOS

- := [Модель, разработанная W3C (World Wide Web Consortium), для представления и обмена структурами организации знаний в семантической сети]
- := [Simple Knowledge Organization System]
- ∈ Система организации знаний
- ⇒ разработчик\*:  
W3C (World Wide Web Consortium)
- ⇒ год создания\*:  
2009
- ⇒ основная парадигма\*:  
декларативное представление знаний
- ⇒ описание\*:  
[SKOS — стандарт для представления систем организации знаний (тезаурусов, классификационных схем, рубрикаторов и других видов контролируемых словарей) в рамках семантической сети.]
- ⇒ описание\*:  
[SKOS построен на основе RDF (Resource Description Framework) и позволяет описывать концепты и отношения между ними в машиночитаемом формате, обеспечивая семантическую совместимость между различными системами организации знаний.]
- ⇒ назначение\*:  
[Стандарт предназначен для организации и связывания знаний в веб-масштабе, создания тезаурусов, классификаций и других систем структурированного представления информации.]
- ⇒ основные конструкции\*:
  - { • [Концепт (skos:Concept)]
  - [Иерархические отношения (skos:broader, skos:narrower)]
  - [Ассоциативные отношения (skos:related)]
  - [Метки (skos:prefLabel, skos:altLabel)]
  - [Схемы концептов (skos:ConceptScheme)]
  - }
- ⇒ поддержка\*:
  - [многоязычность]
  - [иерархическая организация знаний]
  - [перекрестные ссылки между концептами]
- ⇒ применение\*:
  - библиотечные классификации
  - корпоративные тезаурусы
  - онтологии предметных областей
  - семантическая разметка веб-контента
  - системы управления знаниями
  - интеграция данных
- ⇒ примеры использования\*:
  - { • [Библиотечные классификации (УДК, ББК, Дьюи)]
  - [Тезаурусы (EuroVoc, AGROVOC, GEMET)]

- [Системы управления контентом (CMS)]
  - [Семантические поисковые системы]
  - [Государственные классификаторы и стандарты]
- }  
⇒ *реализации\**:  
{• *Apache Jena*  
• *Virtuoso Universal Server*  
• *Stardog*  
• *GraphDB*  
• *SKOS API*  
}  
⇒ *особенности\**:  
{• [простота использования]  
• [совместимость с RDF]  
• [поддержка многоязычия]  
• [гибкость в описании отношений]  
• [интеграция с другими семантическими стандартами]  
}  
⇒ *ограничения\**:  
{• [отсутствие поддержки сложных логических конструкций]  
• [ограниченные возможности для формального вывода]  
• [не предназначен для представления сложных онтологий]  
}  
⇒ *библиографические источники\**:  
{• [Miles A., Bechhofer S. SKOS Simple Knowledge Organization System Reference // W3C Recommendation. — 2009.]  
• [Isaac A. et al. SKOS Simple Knowledge Organization System Primer // W3C Working Group Note. — 2009.]  
• [Allemang D., Hendler J. Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL. — Elsevier, 2011.]  
• [Документация W3C по SKOS. — 2023. — Электронный ресурс. Дата обращения: 27.05.2023. URL: <https://www.w3.org/TR/skos-reference/>]  
}

## **OWL**

- := [Язык описания онтологий для семантической сети]  
:= [OWL (Web Ontology Language)]  
∈ *язык веб-онтологии*  
⇒ *разработчик\**:  
*W3C (World Wide Web Consortium)*  
⇒ *год создания\**:  
*2004*  
⇒ *основная парадигма\**:  
*формальная онтология*  
⇒ *описание\**:  
[OWL — язык описания онтологий для семантической сети, позволяющий формально определять сложные концепты, их свойства и отношения с поддержкой логического вывода.]  
⇒ *описание\**:

- [OWL основан на дескрипционной логике, обеспечивает строгую семантику и совместим с другими стандартами семантического веба (RDF, SPARQL).]
- ⇒ *назначение\**:
- [OWL предназначен для использования приложениями, которым необходимо обрабатывать содержание информации, а не просто предоставление информации людям. OWL способствует большему машинная интерпретируемость веб-контента по сравнению с поддерживаемой XML, RDF и RDF-S путем предоставления дополнительного словаря вместе с формальной семантикой.]
- ⇒ *основные конструкции\**:
- { • [Классы (owl:Class)]
  - [Свойства (owl:ObjectProperty, owl:DatatypeProperty)]
  - [Экземпляры (owl:NamedIndividual, owl:Thing)]
  - [Ограничения (owl:Restriction)]
  - [Аксиомы (owl:Axiom)]
  - [Логические операторы (owl:unionOf, owl:intersectionOf)]
  - }
- ⇒ *профили OWL 2\**:
- { • *OWL 2 EL*  
(для больших онтологий)
  - *OWL 2 QL*  
(для запросов к БД)
  - *OWL 2 RL*  
(для rule-based систем)
  - }
- ⇒ *поддержка\**:
- [логический вывод]
  - [проверка согласованности]
  - [наследование свойств]
  - [транзитивные отношения]
  - [кардинальные ограничения]
- ⇒ *применение\**:
- *промышленные стандарты*
  - *умные системы управления данными*
  - *семантическая интеграция данных*
  - *объяснимый ИИ*
- ⇒ *примеры использования\**:
- { • [биоинформатический проект “Gene Ontology”]
  - [систематизированная номенклатура медицины “SNOMED CT”]
  - [онтология временных концепций “OWL-Time”]
  - [проект по созданию модели машинно-читаемых домашних страниц и социальных сетей “FOAF”]
  - [инициатива по разработке совместимых эталонных онтологий для цифрового производства и промышленности “Industrial Ontologies Foundry”]
  - }
- ⇒ *реализации\**:
- { • *Protégé (редактор онтологий)*
  - *HermiOT (инструмент для OWL 2)*
  - *Apache Jena*
  - *OWLAPI*
  - *Stardog*
  - }

- ⇒ }
  - ⇒ особенности\*:
    - {• [строгая формальная семантика]
    - [поддержка сложных логических конструкций]
    - [возможность автоматизированного вывода]
    - [интеграция с RDF-данными]
    - [поддержка различных профилей]
  - ⇒ }
    - ⇒ ограничения\*:
      - {• [высокий порог вхождения]
      - [вычислительная сложность некоторых конструкций]
      - [избыточность для простых таксономий]
    - ⇒ }
      - ⇒ библиографические источники\*:
        - {• [McGuinness D. L. et al. OWL web ontology language overview //W3C recommendation. – 2004. – Т. 10. – №. 10. – С. 2004.]
        - [OWL 2 Web Ontology Language Document Overview. W3C Recommendation, 2009.]
        - [Hitzler P., et al. OWL 2 Primer. W3C Working Group Note, 2009.]
        - [Antoniou G., van Harmelen F. A Semantic Web Primer. MIT Press, 2008.]
        - [Документация W3C по OWL. — 2023. — URL: <https://www.w3.org/OWL/>]

### **Сравнение SKOS и OWL**

- = {• SKOS (*Simple Knowledge Organization System*)
- OWL (*Web Ontology Language*)
- }
- ⇒ сходства\*:
  - {• [Оба являются стандартами семантического веба и основаны на RDF]
  - [Поддерживают декларативное представление знаний]
  - [Используются для структурирования информации в машиночитаемом формате]
  - [Позволяют определять концепты и отношения между ними]
  - [Интегрируются с другими технологиями семантического веба (SPARQL, RDFS)]
- ⇒ различия\*:
  - {• [SKOS предназначен для организации знаний, OWL – для формального онтологического моделирования]
  - [SKOS поддерживает только простые иерархические отношения (broader/narrower), OWL – сложные логические конструкции]
  - [OWL имеет строгую формальную семантику (дескрипционная логика), SKOS – нет]
  - [OWL поддерживает автоматический логический вывод, SKOS – только навигацию по явным связям]
  - [SKOS оптимален для тезаурусов и классификаций, OWL – для сложных онтологий]
  - [Порог вхождения SKOS значительно ниже, чем OWL]
- }

⇒ *рекомендации по выбору\**:

- { • [SKOS:
    - Библиотечных классификаций (УДК, ББК)
    - Корпоративных тезаурусов и глоссариев
    - Простых таксономий и систем категоризации
    - Многоязычных словарей]
  - [OWL:
    - Сложных предметных онтологий (медицина, инженерия)
    - Систем с автоматическим логическим выводом
    - Моделирования с строгими ограничениями и аксиомами
    - Интеграции разнородных источников данных]
  - [SKOS и OWL:
    - Связать простую таксономию с формальной онтологией
    - Добавить логические правила к существующей классификации
    - Обогащить тезаурус семантическими отношениями]
- }



## 2 ФОРМАЛИЗАЦИЯ И СРАВНИТЕЛЬНЫЙ АНАЛИЗ ИНСТРУМЕНТОВ ДЛЯ РАЗРАБОТКИ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ: AMAZON WEB SERVICES (AWS) AI И GOOGLE CLOUD AI

### *Google Cloud AI*

- :=** [Облачная платформа для разработки и развертывания моделей ИИ от Google]
- :=** [Комплекс сервисов AI/ML от Google Cloud]
- ∈** *облачный сервис искусственного интеллекта*
- ∈** *платформа машинного обучения*
- ∈** *сервис генеративного ИИ*
- ⇒** *разработчик\**:  
*Google LLC*
- ⇒** *год создания\**:  
*2017*
- ⇒** *текущая версия\**:  
*[Vertex AI 1.35]*
- ⇒** *дата обновления\**:  
*[июнь 2025]*
- ⇒** *лицензирование\**:  
  - {**• *Free Tier*
  - *Pay-as-you-go*
  - *Committed Use Discounts*
  - }**
- ⇒** *языки\**:  
  - {**• *Python*
  - *Java*
  - *Go*
  - *Node.js*
  - }**
- ⇒** *назначение\**:  
*[Полный цикл разработки и эксплуатации AI/ML моделей]*
- ⇒** *основные компоненты\**:  
  - {**• *Vertex AI Platform*
    - ⇒** *описание\**:  
*[Унифицированная платформа для ML workflows]*
    - ⇒** *подкомпоненты\**:  
      - {**• *Vertex Workbench (Jupyter ноутбуки)*
      - *Vertex Pipelines (оркестрация)*
      - *Vertex Model Registry*
      - }**
  - *Generative AI Studio*
    - ⇒** *назначение\**:  
*[Работа с генеративными моделями]*
    - ⇒** *модели\**:  
      - {**• *Gemini 1.5 Pro (128K контекст)*
      - *Imagen 2.0 (генерация изображений 1024x1024)*
      - *Codey (код generation)*
      - }**

```

    }
  }
⇒ ключевые особенности*:
  {
    • Гибридные вычисления
      ⇒ описание*:
        [Поддержка Cloud TPU v4 и NVIDIA H100]
      ⇒ производительность*:
        [До 9 EFLOPS в TPU pods]
    • MLOps полного цикла
      ⇒ функции*:
        {
          • Feature Store
          • Model Monitoring
          • Continuous Evaluation
        }
    • Глубокая интеграция
      ⇒ сервисы*:
        {
          • BigQuery ML
          • Looker
          • Google Workspace
        }
      }
⇒ преимущества*:
  {
    • Лидерство в генеративном ИИ
    • Лучшая в отрасли аналитика
    • Автоматическое масштабирование
    • Глобальная сеть (40 регионов)
  }
⇒ недостатки*:
  {
    • Сложность миграции
    • Ограничения кастомизации
    • Высокая стоимость GPU
  }
⇒ библиографические источники*:
  {
    • [Google Cloud AI Documentation [Электронный ресурс] // URL:
      https://cloud.google.com/ai (дата обращения: 27.05.2025)]
    • [Vertex AI: Unified ML Platform [Электронный ресурс] // URL:
      https://cloud.google.com/vertex-ai (дата обращения: 27.05.2025)]
    • [Gemini: Google's Most Capable AI Model [Электронный ресурс] // URL:
      https://blog.google/technology/ai/gemini-google-most-capable-ai-model/ (дата
      обращения: 27.05.2025)]
  }

```

### **AWS AI Services**

```

:= [Комплекс сервисов искусственного интеллекта Amazon Web Services]
:= [Облачная платформа AI/ML от AWS]
∈ облачный сервис ИИ
∈ платформа машинного обучения
∈ сервис генеративного ИИ
⇒ разработчик*:

```

- ⇒ *Amazon.com*
- ⇒ *год создания\**:  
2016
- ⇒ *текущая версия\**:  
[2025.Q2]
- ⇒ *дата обновления\**:  
[май 2025]
- ⇒ *лицензирование\**:
  - {
    - *Free Tier*
    - *On-Demand*
    - *Savings Plans*
    - *Enterprise Agreements*
- ⇒ *поддерживаемые языки\**:
  - {
    - *Python*
    - *Java*
    - *JavaScript*
    - *.NET*
- ⇒ *назначение\**:  
[Корпоративные решения для искусственного интеллекта]
- ⇒ *архитектурные компоненты\**:
  - {
    - *Amazon SageMaker*
      - ⇒ *описание\**:  
[Полнофункциональная ML платформа]
      - ⇒ *подкомпоненты\**:
        - {
          - *SageMaker Studio*
          - *SageMaker Pipelines*
          - *Feature Store*
    - *Bedrock*
      - ⇒ *назначение\**:  
[Генеративный ИИ как сервис]
      - ⇒ *модели\**:
        - {
          - *Claude 3 (Anthropic)*
          - *Stable Diffusion*
          - *AWS Titan*
- ⇒ *ключевые особенности\**:
  - {
    - *Вычислительные мощности*
      - ⇒ *описание\**:  
[EC2 P5 (NVIDIA H100) и Trainium]
      - ⇒ *производительность\**:  
[До 20 EFLOPS в кластерах]
    - *ML промышленного уровня*
      - ⇒ *функции\**:
        - {
          - *Model Registry*
          - *ML Lineage Tracking*

- *Shadow Testing*
- }
- *Гибридные решения*
- ⇒ *сервисы\**:
  - {
    - *Outposts*
    - *Snow Family*
    - *Local Zones*
  - }
- }
- ⇒ *преимущества\**:
  - {
    - *Корпоративная фокусировка*
    - *Самая большая экосистема*
    - *Гибридные сценарии*
    - *Глобальное покрытие (33 региона)*
  - }
- ⇒ *недостатки\**:
  - {
    - *Сложность настройки*
    - *Фрагментированность сервисов*
    - *Долгий выход новых функций*
  - }
- ⇒ *библиографические источники\**:
  - {
    - [AWS Official Documentation [Электронный ресурс] // URL: <https://aws.amazon.com/documentation/> (дата обращения: 27.05.2025)]
    - [Amazon Textract Overview [Электронный ресурс] // URL: <https://aws.amazon.com/textract/> :cite[2] (дата обращения: 27.05.2025)]
    - [LaTeX on AWS Lambda [Электронный ресурс] // URL: <https://github.com/serverlesspub/latex-aws-lambda-layer> :cite[4] (дата обращения: 27.05.2025)]
  - }

### ***Сравнение Google Cloud AI и Amazon Web Services AI***

- = {
  - *Google Cloud AI*
  - *AWS AI Services*
- ⇒ *сходства\**:
  - {
    - [Предоставляют комплексные облачные решения для AI/ML]
    - [Поддерживают полный цикл разработки моделей (от обучения до эксплуатации)]
    - [Обеспечивают инструменты MLOps для управления жизненным циклом моделей]
    - [Предлагают готовые сервисы для компьютерного зрения и NLP]
    - [Интегрированы с экосистемами своих облачных платформ]
    - [Поддерживают популярные ML-фреймворки (TensorFlow, PyTorch)]
    - [Предоставляют мощные вычислительные ресурсы (TPU/GPU)]
    - [Имеют бесплатные тарифные планы для ознакомления]
    - [Обеспечивают enterprise-уровень безопасности]
  - }
- ⇒ *различия\**:
  - {
    - [Google Cloud AI лидирует в генеративном ИИ (Gemini), AWS делает ставку на партнерские модели (Anthropic, Stability AI)]

- [Vertex AI (Google) предлагает более унифицированный интерфейс, SageMaker (AWS) более гибкий но сложный]
- [Google Cloud TPU оптимизированы для TensorFlow, AWS Trainium/Inferentia - для своих сервисов]
- [Интеграция с BigQuery (Google) vs Redshift/Athena (AWS)]
- [Google предлагает более тесную интеграцию с Workspace, AWS - с корпоративными ИТ-системами]
- [AWS имеет более развитые гибридные решения (Outposts, Snow Family)]
- [Google Cloud AI лучше подходит для аналитических задач, AWS - для промышленного ML]
- [Поддержка языков: Google сильнее в Python, AWS - в Java/.NET]
- [Глобальная инфраструктура: 40 регионов у Google vs 33 у AWS]

}

⇒ рекомендации по выбору\*:

- { • [Google Cloud AI рекомендуется для проектов с генеративным ИИ, аналитики данных и интеграции с Google-экосистемой]
- [Google Cloud AI предпочтителен для исследовательских задач и быстрого прототипирования]
- [Google Cloud AI лучше подходит для стартапов и компаний, использующих Google Workspace]
- [AWS AI рекомендуется для корпоративных проектов с требованием гибридных решений]
- [AWS AI предпочтителен для промышленного ML и интеграции с enterprise-системами]
- [AWS AI лучше подходит для Java/.NET разработчиков и сложных ML-пайплайнов]
- [Для медицинских и финансовых проектов: Google - для аналитики, AWS - для compliance]
- [Для IoT решений: AWS с его edge-сервисами (SageMaker Edge)]
- [Для мультимодальных приложений: Google с Gemini]

}

## ЗАКЛЮЧЕНИЕ

В ходе проведённого исследования был осуществлён сравнительный анализ современных технологий разработки интеллектуальных систем: технологий семантических сетей и облачных платформ разработки искусственного интеллекта.

В первом разделе работы были подробно рассмотрены технологии семантических сетей SKOS и OWL, которые составляют теоретическую основу для организации знаний в интеллектуальных системах. Проведён детальный анализ их возможностей, областей применения и ограничений, что позволило сформулировать чёткие критерии выбора между этими стандартами. Результаты анализа были формализованы с помощью SCn-кода.

Во втором разделе представлен подробный сравнительный анализ двух ведущих облачных AI-платформ. Исследование охватило архитектурные особенности, вычислительные возможности, поддерживаемые модели машинного обучения и генеративного ИИ, а также интеграционные аспекты каждой платформы. Особое внимание уделено функциональности и инструментам управления AI-моделями. Информация о каждом инструменте также была формализована на SCn-коде.

Для задач, связанных с организацией знаний, работа предоставляет четкие критерии выбора между SKOS и OWL. В области облачных решений ИИ исследование предлагает систему рекомендаций по выбору платформы в зависимости от типа проекта, требований к интеграции и особенностей решаемых задач. Полученные выводы могут быть использованы при проектировании интеллектуальных систем, сочетающих семантические технологии и современные облачные AI-сервисы.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] Miles A., Bechhofer S. SKOS Simple Knowledge Organization System Reference / Bechhofer S. Miles A. — W3C Recommendation, 2009.
- [2] et al., Isaac A. SKOS Simple Knowledge Organization System Primer / Isaac A. et al. — W3C Working Group Note, 2009.
- [3] Allemang D., Hendler J. Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL / Hendler J. Allemang D. — Elsevier, 2011.
- [4] Документация w3c по skos. — 2023. — Электронный ресурс. Дата обращения: 28.05.2025. <https://www.w3.org/TR/skos-reference/>.
- [5] McGuinness, D. L. Owl web ontology language overview / D. L. McGuinness, F. van Harmelen // W3C Recommendation. — 2004. — Vol. 10, № 10. — P. 2004.
- [6] Owl 2 web ontology language document overview: Tech. rep.: W3C Recommendation, 2009.
- [7] Hitzler, P. Owl 2 primer: Tech. rep. / P. Hitzler, M. Krötzsch, S. Rudolph: W3C Working Group Note, 2009.
- [8] Antoniou, G. A Semantic Web Primer / G. Antoniou, F. van Harmelen. — MIT Press, 2008.
- [9] W3c owl documentation. — 2023. — Accessed: 27.05.2025. <https://www.w3.org/OWL/>.
- [10] Google cloud ai documentation. — 2025. — Accessed: 17.05.2025. <https://cloud.google.com/ai>.
- [11] Vertex ai: Unified ml platform. — 2025. — Accessed: 27.05.2025. <https://cloud.google.com/vertex-ai>.
- [12] Gemini: Google's most capable ai model. — 2025. — Accessed: 27.05.2025. <https://blog.google/technology/ai/gemini-google-most-capable-ai-model/>.
- [13] Aws official documentation. — 2025. — Accessed: 27.05.2025. <https://aws.amazon.com/documentation/>.
- [14] Amazon textract overview. — 2025. — Accessed: 27.05.2025. <https://aws.amazon.com/textract/>.
- [15] Latex on aws lambda. — 2025. — Accessed: 27.05.2025. <https://github.com/serverlesspub/latex-aws-lambda-layer>.