Министерство науки и высшего образования Российской Федерации

ФГБОУ ВО Ульяновский государственный технический университет

Кафедра «Вычислительная техника»

Лабораторная работа №3

«Построение онтологической модели»

Вариант №4

Выполнил студент

группы ИВТИИбд-11

Богданов Р.М.

Преподаватель

Хайрулин И.Д.

Ульяновск, 2025

**1. Структурное описание онтологической модели**

Онтологическая модель описывает музыкальный сервис и формализует ключевые объекты и их взаимосвязи. Основные сущности или классы модели включают:

* **Исполнители (Artist):** объекты, тех или иных музыкальных исполнителей в первую очередь групп.
* **Альбомы (Album):** созданная исполнителем коллекция треков, часто завязанная на общей тематике.
* **Треки (Track):** музыкальные произведения, созданные исполнителями, попадающие в плейлисты, и часто являющиеся частью альбомов тех или иных исполнителей.
* **Плейлисты (Playlist):**. коллекция музыкальных произведений часто имеющая общий жанр или настроение.

Для реализации модели использованы два инструмента:

1. **Protégé** — применяется для построения OWL-онтологии, добавления аннотаций и формулировки логических правил с использованием SWRL.
2. **Neo4j** — используется для создания графовой модели с узлами и связями, где логические зависимости реализуются с помощью языка запросов Cypher.

**2. Перечень классов, отношений и аксиом**

Онтологическая модель включает несколько типов элементов: классы (сущности), свойства данных и свойства объектов, а также логические правила (аксиомы), описывающие взаимосвязи между объектами.

**2.1 Классы (Classes)**

Модель содержит следующие основные классы:

* **Исполнители (Artist):** представляет литературные произведения, которые могут быть написаны авторами и опубликованы издательствами.
* **Альбомы (Album):** лица, создающие книги и участвующие в процессе публикации.
* **Треки (Track):** пользователи библиотеки, которые берут книги на время и взаимодействуют с другими сущностями через взятые книги.
* **Плейлисты (Playlist):** организации, выпускающие книги, связанные с авторами и читателями через свои публикации.

Каждый класс является фундаментальной единицей модели и служит основой для определения свойств и отношений.

**2.2 Свойства данных (Data Properties)**

|  |  |  |
| --- | --- | --- |
| **Свойство** | **Принадлежит классу** | **Тип данных** |
| Имя | Все | Строка |
| Год выпуска | Альбом | Число |
| Длительность | Трек | Число секунд |
| Настроение | Трек, Плейст | Строка |
| Жанр | Альбом, Трек, Исполнитель | Строка |

Таблица – Свойства данных Protégé

|  |  |  |
| --- | --- | --- |
| **Свойство** | **Принадлежит классу** | **Тип данных** |
| Name(title) | Все | Строка |
| Release data | Альбом | Дата |
| Duration | Трек | Число секунд |
| Mood | Трек, Плейст | Строка |
| Genre | Альбом, Трек, Исполнитель | Строка |

Таблица – Свойства данных Neo4j

**2.3 Свойства объектов (Object Properties)**

|  |  |
| --- | --- |
| **Свойство** | **Описание** |
| Входит в альбом | Трек в альбом |
| Входит в плейст | Трек в плейст |
| Создает | Исполнитель трек |
| Создал альбом | Исполнитель альбом |

Таблица – Отношения объектов Protégé

|  |  |
| --- | --- |
| **Отношение** | **Описание** |
| CONTAINS | Альбом трек |
| INCLUDES | Плейст трек |
| PREFORMS IN | Исполнитель альбом |

Таблица – Связи данных Neo4j

**2.4 Аксиомы (SWRL-правила) для Protégé**

Если жанр русский рок плейст русский рок:

untitled-ontology-4:Трек(?t) ^ untitled-ontology-4:Жанр(?t, "Русский рок") -> untitled-ontology-4:входит\_в\_плейст(untitled-ontology-4:Русский\_рок, ?t)

Если черный альбом то кино:

untitled-ontology-4:Трек(?t) ^ untitled-ontology-4:входит\_в\_альбом(?t, untitled-ontology-4:Черный\_альбом) -> untitled-ontology-4:создает(untitled-ontology-4:КИНО, ?t)

# 2.5 Аналог аксиом для Neo4j (Cypher-запросы)

1. Выводит если плейлист содержит больше 2 треков с жанром русский рок то его тематика русский рок**:**

MATCH (p:Playlist)-[:INCLUDES]->(t:Track)

WHERE t.genre = "Русский рок"

WITH p, COUNT(t) AS rus\_rock\_tracks

WHERE rus\_rock\_tracks >= 2

RETURN p.name AS thematic\_playlist, rus\_rock\_tracks AS matching\_tracks;

1. Выводит альбомы где число треков с продолжительностью меньше 4 минут:

MATCH (alb:Album)-[:CONTAINS]->(t:Track)

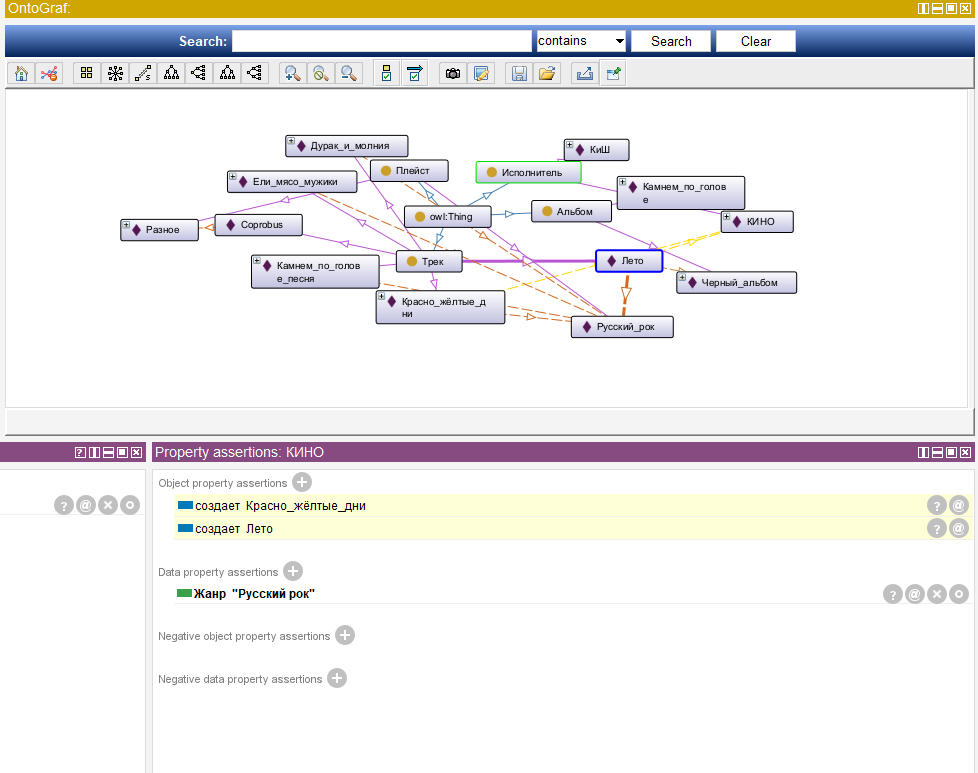
WHERE t.duration < 240

WITH alb, COUNT(t) AS short\_tracks

WHERE short\_tracks >= 2

RETURN alb.title AS album, short\_tracks AS tracks\_under\_4min;

# Реализация в Protégé

Рисунок демонстрация иерархии и правила “Если черный альбом то кино”.

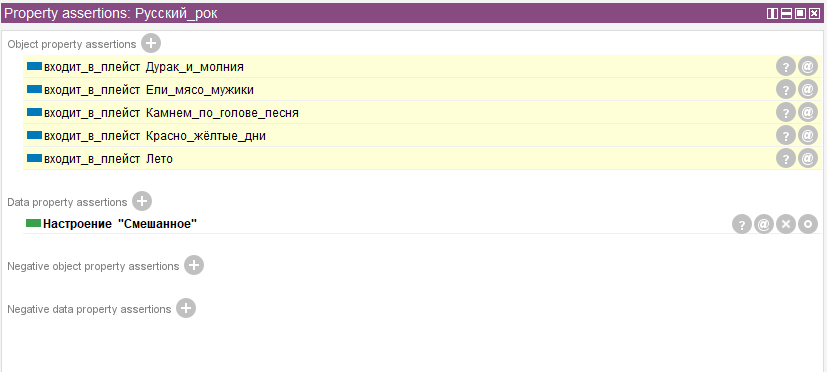


Рисунок правила “ Если жанр русский рок плейст русский рок ”.

# 4. Реализация в Neo4j

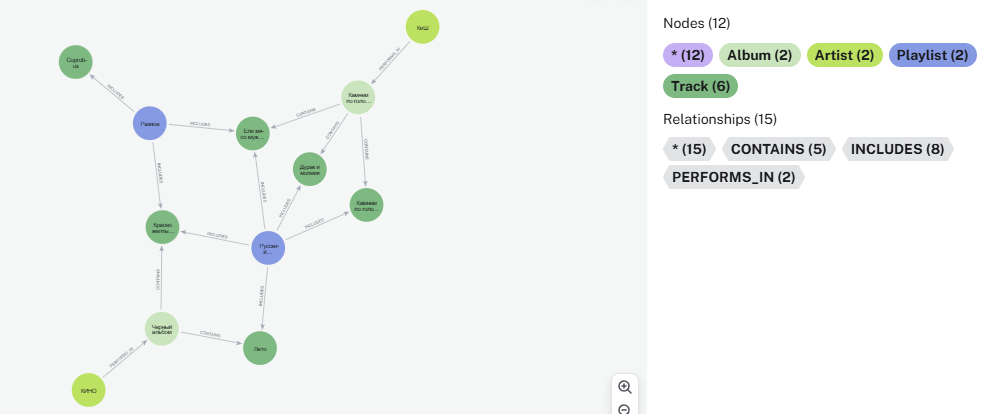


Рисунок реализация

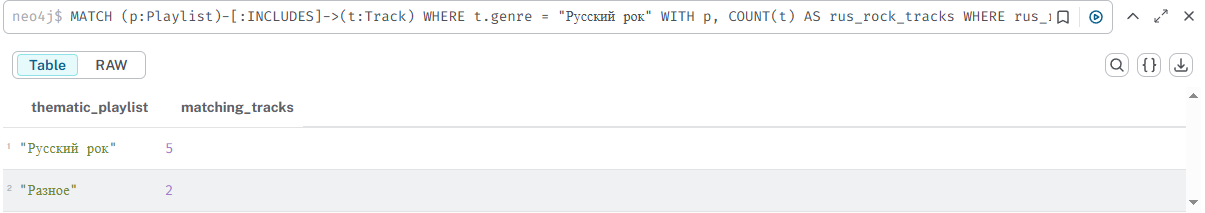


Рисунок запрос плейлист содержит больше 2 треков с жанром русский рок то его тематика русский рок.

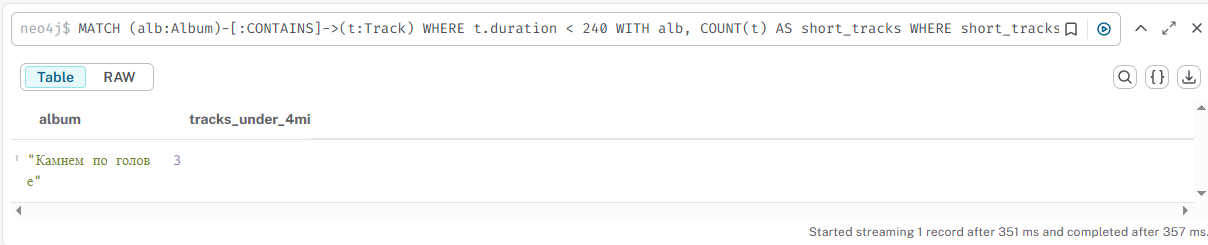


Рисунок запрос альбомы где число треков с продолжительностью меньше 4 минут.

**5. Вывод**

В работе создана онтологическая модель музыкального сервиса, объединяющая исполнителей, альбомы, треки и плейлисты так же были созданы SWRL-правила и Cypher-запросы, позволяющие автоматически выводить обратные и косвенные связи.