
КАФЕДРА

ОТЧЕТ ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

должность, уч. степень, звание

подпись, дата

инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №1

Разработка концепта требований по
курсу: Разработка и анализ требований

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ гр. №

подпись, дата

Столяров Н.С.

инициалы, фамилия

Санкт-Петербург 2025

Содержание	Задание:	3
1. Введение и постановка задачи		3

2. Основная идея концепта.....	3
3. Функциональные требования.....	4
3.1 Модуль сбора и преобразования данных:.....	4
3.2 Модуль многоуровневой трансформации:.....	4
3.3 Онтологическая модель процессов:.....	5
3.4 Модуль хранения и визуализации:.....	5
3.5 Интеграция и обмен данными:.....	5
4. Нефункциональные требования.....	5
5. Технологические аспекты.....	6
6. Сценарии использования и практическая значимость.....	7
6.1 Мониторинг технических объектов:.....	7
6.2 Динамическая адаптация алгоритмов:.....	7
6.3 Интеграция с аналитическими платформами:.....	7
7. Диаграмма.....	8
8. Заключение.....	9

Задание:

Разработать концепт требований к системе хранения онтологической модели структур данных.

2. Введение и постановка задачи

Современные технические объекты, включая промышленные установки, транспортные средства и системы мониторинга, производят огромные массивы данных в виде временных рядов. Эти данные, поступающие от различных датчиков, часто оказываются неоднородными, зашумленными и непригодными для непосредственного анализа. В связи с этим возникает необходимость их предварительной обработки, преобразования и структурирования для последующего анализа, прогнозирования и принятия решений.

Цель данной работы — разработать концептуальную модель системы, выполняющей многоуровневую трансформацию временных рядов на основе онтологического подхода. Такая система не только преобразует исходные данные в удобный для анализа формат, но и обеспечивает прозрачное документирование каждого этапа обработки с помощью формализованной онтологии. Кроме того, интеграция с графовой базой данных позволяет эффективно хранить сложные взаимосвязи между этапами трансформации, алгоритмами, методами оценки и параметрами данных.

3. Основная идея концепта

Система включает несколько модулей, каждый из которых отвечает за определённый этап обработки временных рядов:

- Сбор и предварительная обработка данных: система принимает потоки данных с датчиков, выполняет первичную проверку, очистку и агрегацию, устраняя шумы и заполняя пропуски.
- Многоуровневая трансформация данных: данные проходят через последовательность этапов, включая фильтрацию, сглаживание, аппроксимацию, интерполяцию, агрегацию и применение статистических моделей.
- Онтологическое описание процессов: каждый этап обработки формализуется с помощью онтологической модели, включающей классы для этапов трансформации, алгоритмов обработки, методов оценки и связей между данными.
- Хранение и управление знаниями: онтологическая модель сохраняется в графовой базе данных, что обеспечивает быстрый поиск, визуализацию связей и динамическое обновление структуры.

- **Интеграция с внешними системами:** система предоставляет REST API для взаимодействия с другими платформами, что важно для комплексного мониторинга и поддержки принятия решений.

4. Функциональные требования

Модуль сбора и предобработки: приём, очистка и агрегация данных.

Модуль трансформации: многоэтапная обработка с возможностью выбора алгоритмов.

Онтологическая модель: формализация этапов, алгоритмов, методов оценки и связей между ними.

Модуль хранения и визуализации: использование графовой БД, визуализация цепочек обработки, поддержка версионности.

Интеграция: REST API, экспорт в стандартных форматах, динамическое обновление параметров.

5. Нефункциональные требования

- **Производительность:** обработка больших объёмов данных с минимальными задержками.
- **Масштабируемость:** возможность расширения для работы с растущими объёмами данных.
- **Надёжность:** резервное копирование и механизмы восстановления.
- **Безопасность:** шифрование данных, аутентификация и авторизация.
- **Гибкость:** возможность изменения алгоритмов и структуры онтологии без значительных доработок.
- **Документированность:** детальное описание процессов для удобства поддержки.

6. Технологические аспекты

- **Онтологии:** RDF/OWL.
- **Графовая БД:** Neo4j или GraphDB.

- Обработка данных: Python/Java с использованием библиотек для временных рядов и ML.
- API: RESTful для интеграции.
- Визуализация: веб-интерфейс для отображения графовых связей.

7. Сценарии использования и практическая значимость

Мониторинг технических объектов: выявление аномалий, прогнозирование отказов.

Динамическая адаптация алгоритмов: быстрое изменение правил обработки через онтологию.

Интеграция с аналитическими платформами: передача данных в BI-системы и ML-решения.

8. Диаграмма

Диаграмма иллюстрирует структуру системы, предназначенной для многоуровневой трансформации временных рядов технических объектов. Основные компоненты:

- Модуль сбора и предобработки данных получает и очищает данные с датчиков.
- Модуль многоуровневой трансформации данных выполняет последовательную обработку (фильтрация, аппроксимация, агрегация).
- Онтологическая модель процессов формализует этапы обработки, описывает используемые алгоритмы и методы оценки.
- Графовая база данных хранит онтологическую модель, обеспечивая быстрый поиск и визуализацию взаимосвязей.
- REST API обеспечивает интеграцию с внешними системами, обмен данными и обратную связь для обновления процессов обработки.

Такая архитектура позволяет гибко адаптировать систему под изменяющиеся условия и требования, обеспечивая высокую производительность и масштабируемость.



Рисунок 1 – Диаграмма структуры концептуальной системы

9. Заключение

Концепция объединяет онтологический подход и графовые базы данных, обеспечивая структурированную, гибкую и масштабируемую систему для обработки временных рядов. Это делает её применимой в промышленности и научных исследованиях.