



18. Volchkov V. P., Poborchaya N. Ye. Predstavlenie sluchainikh protsessov vektornoj rekurrentnoi tsirkulyantnoi mode-lyu vtorogo poryadka [Representation of random processes by a second-order vector recurrent circulant model]. *Zhurnal radioelektroniki (elektronnyi zhurnal IRE RAN)* [Journal of Radio Electronics (Electronic Journal of IRE RAS)]. 2013, no. 12. Available at: <http://jre.cplire.ru/jre/dec13/14/text.pdf>.

19. Volchkov V. P., Poborchaya N. Ye., Shloma A. M. Approksimatsiya, filtratsiya i sglazhivanie sluchainikh signalov s pomoshchyu rekurrentnikh tsirkulyantnikh modelei skolzyashchego okna [Approximation, filtering and smoothing of random signals using recurrent circulant sliding window models]. *Nauchnie vedomosti Belgorodskogo gosudarstvennogo universiteta. Ser. Istoriya. Politologiya. Ekonomika. Informatika* [Scientific Bulletin of Belgorod State University. Series: History. Political Science. Economics. Computer Science]. 2013, vol. 28/1, no. 22 (165), pp. 135–143.

© К. Д. Яксубаев, И. В. Аксютин

**Ссылка для цитирования:**

Яксубаев К. Д., Аксютин И. В. Уравнение Кардано – Тарталья и матрицы-циркулянты // Инженерно-строительный вестник Прикаспия : научно-технический журнал / Астраханский государственный архитектурно-строительный университет. Астрахань : ГБОУ АО ВО «АГАСУ», 2026. № 1 (55). С. 100–104.

УДК 004.9 (075.8)

DOI 10.52684/2312-3702-2026-55-1-104-111

**ИНТЕЛЛЕКТУАЛЬНАЯ СИСТЕМА ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ  
ДЛЯ УПРАВЛЕНИЯ МАШИННЫМИ НОСИТЕЛЯМИ ИНФОРМАЦИИ  
НА ОСНОВЕ ЧЕЛОВЕКО-МАШИННОГО КОЛЛЕКТИВНОГО ИНТЕЛЛЕКТА**

**О. В. Кудрявцева, Е. М. Бялецкая, М. А. Кудрявцева**

**Кудрявцева Ольга Витальевна**, старший преподаватель кафедры экономики строительства, Астраханский государственный архитектурно-строительный университет; аспирант, Астраханский государственный технический университет, г. Астрахань, Российская Федерация; e-mail: kudryavtzevaov@mail.ru;

**Бялецкая Елена Михайловна**, кандидат технических наук, доцент кафедры «Информатика, математика и общегуманитарные науки», Новороссийский филиал Финансового университета при Правительстве Российской Федерации, г. Новороссийск, Российская Федерация;

**Кудрявцева Мария Алексеевна**, студент, Астраханский государственный архитектурно-строительный университет, г. Астрахань, Российская Федерация

В условиях цифровой трансформации организациям необходимо эффективно управлять данными. Традиционные системы учета не справляются со сложными задачами, требующими экспертной оценки. Данное исследование предлагает интеллектуальную систему поддержки принятия решений, объединяющую надежность PostgreSQL, гибкость Python и возможности коллективного интеллекта человека и машины. В статье представлена концепция и архитектура системы автоматизированного учета машинных носителей информации. Описаны математическая модель, реляционная схема базы данных и алгоритмы взаимодействия компонентов. Реализация на Python с использованием psycopg2 обеспечивает эффективную и безопасную работу с СУБД. Система предназначена для автоматизации учета и организации совместной работы экспертов и программных агентов при решении задач информационной безопасности и оптимизации ресурсов. Разработанная система является масштабируемой и может быть адаптирована для организаций различного профиля.

**Ключевые слова:** система поддержки принятия решений, человеко-машинный коллективный интеллект, управление машинными носителями информации, PostgreSQL, Python, реляционная база данных, краудсорсинг, самоорганизация.

**AN INTELLIGENT DECISION SUPPORT SYSTEM  
FOR MANAGING MACHINE-BASED INFORMATION CARRIERS  
BASED ON HUMAN-MACHINE COLLECTIVE INTELLIGENCE**

**O. V. Kudryavtseva, Ye. M. Byaletsckaya, M. A. Kudryavtseva**

**Kudryavtseva Olga Vitalyevna**, Senior Lecturer of Construction Economics Department, Astrakhan State University of Architecture and Civil Engineering; post-graduate student, Astrakhan State Technical University, Astrakhan, Russian Federation; e-mail: kudryavtzevaov@mail.ru;

**Byaletsckaya Yelena Mikhaylovna**, Candidate of Technical Sciences, Associate Professor of Computer Science, Mathematics, and General Humanities Department, Novorossiysk Branch of the Financial University under the Government of the Russian Federation, Novorossiysk, Russian Federation;

**Kudryavtseva Mariya Alekseyevna**, student, Astrakhan State University of Architecture and Civil Engineering, Astrakhan, Russian Federation

In the context of digital transformation, organizations need to manage their data effectively. Traditional accounting systems are not equipped to handle complex tasks that require expert evaluation. This research proposes an intelligent decision support system that combines the reliability of PostgreSQL, the flexibility of Python, and the capabilities of human-machine collective intelligence. The article presents the concept and architecture of an automated accounting system for machine-readable media. It describes a mathematical model, a relational database schema, and algorithms for component interaction. The implementation in Python using psycopg2 ensures efficient and secure operation with the database management system. The system is designed

to automate the accounting and organization of collaborative work of experts and software agents in solving information security and resource optimization tasks. The developed system is scalable and can be adapted for organizations of various profiles.

**Keywords:** *decision support system, human-machine collective intelligence, machine-readable media (MRM) management, PostgreSQL, Python, relational database, crowdsourcing, self-organization.*

### Введение (Introduction)

В эпоху больших данных и повсеместной цифровизации машинные носители информации (МНИ) остаются ключевым физическим активом, от управления которым зависят информационная безопасность и непрерывность бизнес-процессов [1]. Крупные организации сталкиваются со значительными трудностями в учете, отслеживании перемещения и контроле состояния тысяч единиц МНИ, начиная от жестких дисков и заканчивая съемными носителями [2]. Существующие подходы к автоматизации учета, как правило, основаны на жестко заданных регламентах и workflow, что делает их негибкими и неспособными адаптироваться к новым, слабоформализованным вызовам, таким как расследование инцидентов или стратегическое планирование ресурсов [3].

Параллельно с этим, благодаря развитию информационно-коммуникационных технологий и искусственного интеллекта (ИИ), все более широкое применение находят системы человеко-машинных вычислений [4]. Однако в большинстве таких систем человек рассматривается лишь как «вычислительное устройство» для выполнения узкоспециализированных задач (например, классификации изображений), что игнорирует его творческий потенциал и способности к самоорганизации и решению комплексных проблем [5].

Целью данного исследования является преодоление указанных ограничений путем разработки и реализации интеллектуальной системы поддержки принятия решений (СППР) для управления МНИ. Данная система интегрирует три ключевых компонента:

- надежную реляционную базу данных (PostgreSQL) для структурированного хранения всей информации;
- гибкий язык программирования (Python) для реализации прикладной логики, аналитических модулей и интерфейсов взаимодействия;
- концепцию человеко-машинного коллективного интеллекта (ЧМКИ), которая позволяет объединить экспертизу людей-специалистов и вычислительную мощь программных агентов для решения слабоструктурированных задач.

Для достижения поставленной цели в работе решаются следующие задачи:

- разработка концептуальной модели и архитектуры СППР на основе ЧМКИ;
- проектирование и реализация реляционной схемы базы данных PostgreSQL, обеспечивающей целостность и структурированность данных об МНИ;
- обеспечение эффективного и безопасного взаимодействия между Python-приложением и СУБД;
- демонстрация практической применимости системы на примере решения конкретных задач управления МНИ.

### Метод (Methods)

Достижение поставленной в работе цели – разработки интеллектуальной СППР для управления МНИ – потребовало применения комплекса взаимодополняющих методов, охватывающих как теоретическое моделирование, так и практическую реализацию программного обеспечения. Методологическую основу исследования составили теория баз данных, реляционная алгебра, объектно-ориентированное программирование, а также концептуальные положения теории принятия решений и коллективного интеллекта.

#### 1. Концептуальное моделирование и архитектурный синтез

На начальном этапе для формализации предметной области и требований к системе использовались методы системного анализа и концептуального моделирования. Основой для разработки архитектуры послужила ранее предложенная концептуальная модель поддержки принятия решений на основе человеко-машинного коллективного интеллекта (ЧМКИ) [3]. В рамках этой модели были выделены ключевые абстракции («Проблема», «Коллектив», «Модель процесса») и определены отношения между ними.

«Проблема» – это вопрос, требующий разрешения в контексте управления МНИ (например, «Расследование инцидента утери носителя», «Оптимизация закупок новых накопителей») [18].

«Коллектив» – это группа участников, объединяющая экспертов-людей (администраторов, аналитиков безопасности, руководителей подразделений) и программные сервисы (алгоритмы анализа данных, генераторы отчетов, предиктивные модели).

Модель процесса описывает процедуру поддержки принятия решений, включая сбор информации, разработку и оценку альтернатив. Ключевой особенностью является гибкость этой модели – она может уточняться и адаптироваться самими участниками коллектива в ходе работы [6].

Взаимодействие этих понятий осуществляется в специальной Среде, которая обеспечивает семантическую интероперабельность (взаимопонимание между людьми и машинами) и координацию совместной деятельности.

Метод архитектурного синтеза позволил декомпозировать систему на функциональные уровни (уровень данных, уровень логики и сервисов, уровень коллективного интеллекта) и определить интерфейсы взаимодействия между ними. Архитектура предлагаемой системы, визуализированная на рисунке 1, реализует данную концептуальную модель и состоит из трех основных уровней:

1. Уровень данных (Data Layer), его основу составляет реляционная база данных PostgreSQL, отвечающая за надежное и структурированное хранение всей информации [7]. Она выступает в роли формализованного источника знаний для всей системы.

2. Уровень логики и сервисов (Logic & Services Layer) реализован на языке Python [10]. Он включает в себя:

- базовую логику приложения для выполнения CRUD-операций (Create, Read, Update, Delete) с данными об МНИ;
- интеллектуальные сервисы (Программные агенты), которые выполняют аналитические функции: выявление аномалий, прогнозирование потребностей, автоматическое формирование отчетов [12];
- модуль координации, который управляет взаимодействием между участниками коллектива,

распределяет задачи и обеспечивает выполнение модели процесса [8].

3. Уровень коллективного интеллекта (Collective Intelligence Layer) – это интерфейс для взаимодействия участников. Человек-эксперт через пользовательский интерфейс получает задачи, анализирует данные, вносит предложения по изменению процесса. Программный сервис, в свою очередь, может автоматически генерировать гипотезы или запрашивать у человека уточняющую информацию для своей работы [9].

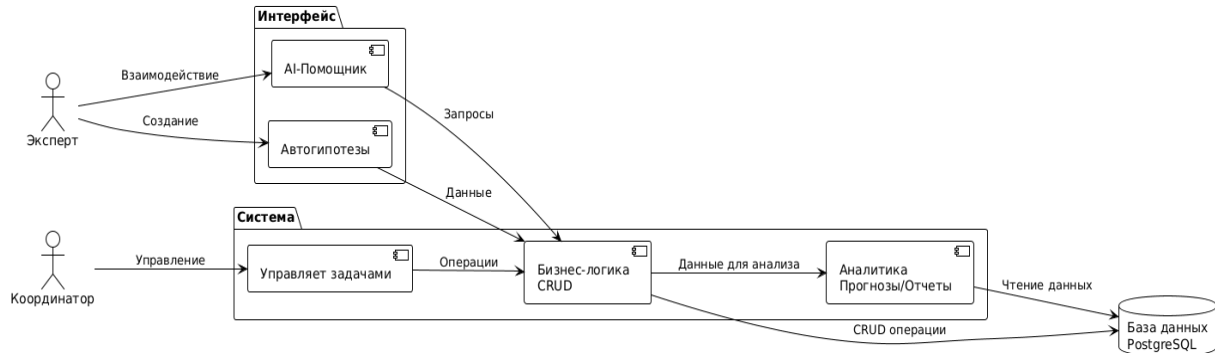


Рис. 1. Архитектура интеллектуальной системы поддержки принятия решений для управления МНИ (иллюстрация авторов <https://editor.plantuml.com/>)

Fig. 1. Architecture of an intelligent decision support system for managing MNI (illustration by the authors <https://editor.plantuml.com/>)

Такой подход обеспечил модульность, масштабируемость и гибкость разрабатываемого решения.

## 2. Математическое и реляционное моделирование данных

Для обеспечения структурированного, целостного и непротиворечивого хранения информации об МНИ и сопутствующих сущностях был применен математический аппарат реляционной алгебры. Формальная модель данных была представлена в виде кортежа (формула 1):

$$DB = (S, R, C), \quad (1)$$

Где  $S = \{S_1, S_2, \dots, S_n\}$  – множество схем отношений (таблиц);  $R = \{r_1, r_2, \dots, r_m\}$  – множество функциональных зависимостей;  $C = \{c_1, c_2, \dots, c_k\}$  – множество ограничений целостности.

Модель включает следующие ключевые наборы данных, реализованные в виде таблиц в PostgreSQL [16, 17]:

- базовые сущности управления доступом, т.е. users (пользователи), roles (роли), user\_roles (связь пользователей и ролей), table\_permissions (разрешения на таблицы);
- журналирование, т.е. logs (логи действий пользователей);
- справочники: departments (подразделения), media\_types (типы носителей), personal\_marks (личные пометки), confidentiality\_levels (грифы конфиденциальности);
- основные данные об МНИ, т.е. набор таблиц (AUP, BPO, FRG и др.), содержащих идентичную структуру атрибутов для описания каждой еди-

ницы носителя (уникальный номер, подразделение, тип, ФИО ответственного, дата выдачи, уровень конфиденциальности и пр.) [13].

Отношения между таблицами устанавливаются с помощью внешних ключей и определяют связи «один-ко-многим», например: departments 1:N AUP (в одном подразделении может быть много МНИ), users 1:N logs (один пользователь может совершить много действий) [14].

На основе этой модели методом нормализации (последовательный перевод в третью нормальную форму) была спроектирована реляционная схема базы данных, минимизирующая избыточность и исключающая аномалии обновления. Логическое проектирование завершилось построением ER-диаграммы, которая наглядно отображает сущности предметной области (пользователи, роли, типы носителей, подразделения, журналы, непосредственно единицы МНИ) и связи между ними. Разработанная реляционная схема, представленная в виде ER-диаграммы на рисунке 2, обеспечивает нормализованную структуру, минимизирует избыточность данных и гарантирует целостность за счет системы ограничений (PRIMARY KEY, FOREIGN KEY, UNIQUE, NOT NULL).

## 3. Реализация взаимодействия с СУБД на языке Python

Физическая реализация спроектированной базы данных была выполнена с использованием реляционной СУБД PostgreSQL, выбор которой обусловлен ее надежностью, поддержкой целостности данных на высоком уровне и развитыми механизмами разграничения доступа. Для организации взаимодействия

между прикладным программным обеспечением и СУБД был выбран язык Python [19–20]. Основным методом интеграции явилось использование библиотеки `psycopg2` – зрелого и полнофункционального адаптера, реализующего протокол взаимодействия с PostgreSQL на уровне C-библиотеки `libpq`. Ключевыми аспектами программной реализации стали:

- применение параметризованных запросов, т.е. данный метод обеспечивает защиту от атак путем внедрения SQL-кода (SQL injection), отделяя структуру запроса от пользовательских данных;
- оптимизация запросов, т. е. для повышения производительности применялся анализ планов выполнения запросов с использованием команды

EXPLAIN, а также создание индексов по часто используемым в предикатах полям;

- управление транзакциями и соединениями, т. е. для обеспечения атомарности операций и предотвращения утечек ресурсов использовались контекстные менеджеры Python, гарантирующие корректное завершение транзакций и закрытие соединений с базой данных даже в случае возникновения исключительных ситуаций.

SQL-код для создания ключевых таблиц представлен на рисунке 3.

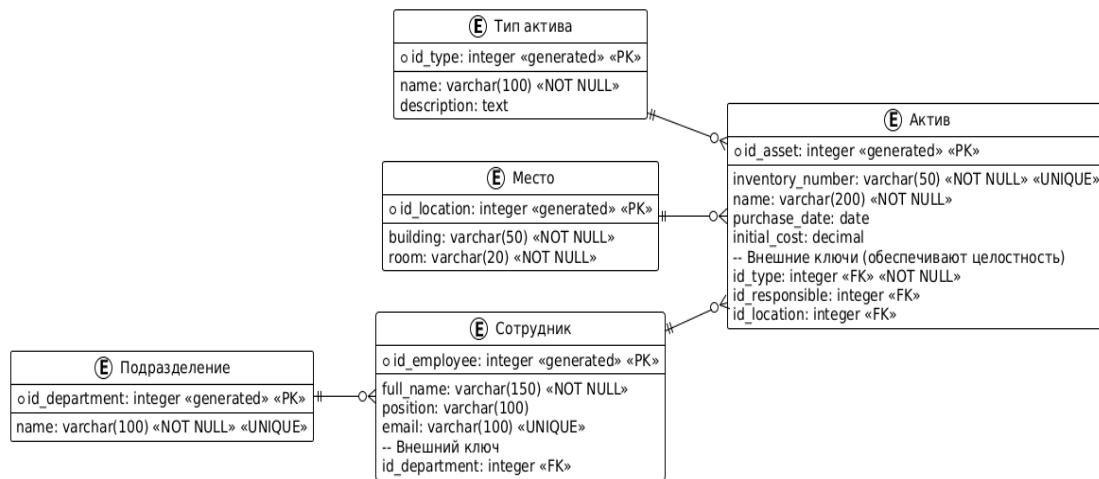


Рис. 2. ER-диаграмма реляционной базы данных системы учета МНИ (иллюстрация авторов <https://editor.plantuml.com/>)

Fig. 2. ER diagram of the relational database of the MNI accounting system (illustration by the authors <https://editor.plantuml.com/>)

```
# ТАБЛИЦА: user_roles (связь пользователей и ролей)
# =====
CREATE_USER_ROLES = """
CREATE TABLE IF NOT EXISTS user_roles (
    user_id INTEGER REFERENCES users(id) ON DELETE CASCADE,
    role_id INTEGER REFERENCES roles(id) ON DELETE CASCADE,
    PRIMARY KEY (user_id, role_id)
);
```

Рис. 3 Реализация взаимодействия Python и PostgreSQL (иллюстрация авторов)  
Fig. 3 Implementation of Python and PostgreSQL interaction (illustration by the authors)

Связующим звеном между реляционной базой данных и интеллектуальными сервисами является язык программирования Python [10]. Для обеспечения эффективного, безопасного и надежного взаимодействия был реализован алгоритм, включающий несколько ключевых этапов.

1. Установление соединения. Используется библиотека `psycopg2` (или `asynprg` для асинхронного подхода). Параметры подключения (хост, порт, пользователь, пароль, имя базы данных) выносятся в отдельный конфигурационный файл (`config_db.py`) для безопасности и удобства управления.

2. Защита от SQL-инъекций. Для этого применяется параметризация запросов, при которой данные отделяются от логики SQL-команды.

3. Оптимизация запросов. Рекомендуются использовать конкретные перечни столбцов вместо `SELECT *`, анализировать план выполнения запроса с помощью EXPLAIN, а также создавать индексы для ускорения поиска по часто используемым полям.

4. Управление соединениями и транзакциями. Для предотвращения утечек ресурсов и обеспечения атомарности операций используются контекстные менеджеры.



Результат успешного выполнения запроса данных из базы и готовность системы к их дальнейшей обработке. (рис. 4) демонстрирует корректное извлечение

Листинг 1. Фрагмент кода Python для выполнения запроса к PostgreSQL и вывода результатов в структурированном виде.

```
import psycopg2
from psycopg2 import OperationalError, sql
from tabulate import tabulate # для форматированного вывода
def execute_query_and_display(db_config: dict, query: str, params: tuple = None) -> None:
    Устанавливает соединение с БД, выполняет параметризованный запрос
    и выводит результат в табличном виде.
    Args:
        db_config: Словарь с параметрами подключения (host, database, user, password, port).
        query: SQL-запрос (может содержать плейсхолдеры %s).
        params: Кортеж параметров для запроса (для защиты от SQL-инъекций).
    connection = None
    cursor = None
    try:
        # 1. Установка соединения с СУБД
        print("[ИНФОРМАЦИЯ] Установка соединения с PostgreSQL...")
        connection = psycopg2.connect(**db_config)
        cursor = connection.cursor()
        print("[ИНФОРМАЦИЯ] Соединение успешно установлено.")
        # 2. Выполнение параметризованного запроса
        print(f"[ИНФОРМАЦИЯ] Выполнение запроса: {query}")
        cursor.execute(query, params)
        # 3. Извлечение результата
        result_set = cursor.fetchall()
        # Получение имен столбцов для заголовка таблицы
        column_names = [desc[0] for desc in cursor.description]
        # 4. Вывод результата в форматированном виде
        if result_set:
            print("[РЕЗУЛЬТАТ] Данные успешно получены:")
            print(tabulate(result_set, headers=column_names, tablefmt="grid"))
        else:
            print("[РЕЗУЛЬТАТ] Запрос не вернул данных.")
    except OperationalError as e:
        print(f"[ОШИБКА] Проблема подключения или выполнения запроса: {e}")
    except Exception as e:
        print(f"[ОШИБКА] Непредвиденная ошибка: {e}")
```

Рис. 4. Результат выполнения запроса к БД из Python-приложения (иллюстрация авторов)  
Fig. 4. The result of a database query from a Python application (illustration by the authors)

#### 4. Разработка алгоритмов человеко-машинного взаимодействия

Реализация концепции ЧМКИ потребовала разработки алгоритмов, координирующих работу людей-экспертов и программных агентов в рамках

решения конкретных задач управления МНИ. В основе этих алгоритмов лежит сценарий, инициируемый постановкой «Проблемы». Далее Модуль координации (часть уровня логики), реализованный на Python, выполняет следующие функции:

- динамическое формирование коллектива на основе семантики проблемы и профилей участников (навыки, компетенции, доступные программные сервисы);

- управление моделью процесса, т. е. предоставление участникам гибкой, адаптируемой модели процесса, которая может дополняться и изменяться в ходе решения задачи;

- обеспечение семантической интероперабельности, т. е. преобразование данных из реляционного хранилища в формат, пригодный для восприятия человеком, и интерпретация действий человека для выполнения соответствующих операций системой.

Предложенная совокупность методов позволила создать не просто информационную систему для учета, а интеллектуальную среду, способную адаптироваться к изменяющимся условиям и привлекать коллективный интеллект для решения сложных, слабоструктурированных задач управления МНИ.

### Результаты и обсуждение (Results and Discussion)

Разработанная система была протестирована на решении ряда практических задач управления МНИ. Рассмотрим два сценария, демонстрирующих синергию между реляционной базой данных, Python-логикой и человеко-машинным коллективным интеллектом [15].

#### Сценарий 1. Расследование инцидента утери МНИ

Проблема: зафиксирована утрата флеш-накопителя с конфиденциальной информацией.

Процесс:

1. Инициация. Система автоматически создает проблему в среде. Модель процесса включает этапы: «Сбор исходных данных», «Анализ логов», «Опрос сотрудников», «Формирование заключения».

2. Сбор данных. Программный сервис (Python-агент) автоматически запрашивает из БД (logs, aur) всю историю перемещений данного носителя и формирует первичный отчет.

3. Коллективный анализ. Эксперт по безопасности (человек) изучает отчет. Через среду он назначает задание сотруднику службы безопасности (другой человек) провести опрос в подразделении. Одновременно с этим, другой программный сервис анализирует логи доступа к смежным системам на предмет аномалий.

4. Адаптация процесса. В ходе расследования эксперты могут добавить в модель процесса новый этап «Проверка камер видеонаблюдения», назначив ответственным другого сотрудника.

5. Формирование решения. Все артефакты (отчеты, результаты опроса, анализ аномалий) агрегируются системой. Коллектив совместно анализирует их и формулирует итоговое заключение для ЛПР.

#### Сценарий 2. Оптимизация закупок МНИ

Проблема: требуется снизить затраты на закупку новых накопителей без ущерба для операционной деятельности.

Процесс:

1. Инициация. Менеджер по закупкам формулирует проблему.

2. Анализ данных. Python-сервис с использованием библиотек pandas и scikit-learn анализирует данные из таблиц AUP, BPO и т.д., строит прогноз выхода из строя носителей и оценивает будущую потребность.

3. Генерация альтернатив. На основе прогноза система (программный агент) предлагает несколько альтернативных стратегий закупок (например, «оптовые закупки раз в квартал», «использование арендованных носителей»).

4. Экспертная оценка. Менеджер по закупкам, IT-директор и финансовый аналитик (люди) через среду обсуждают предложенные альтернативы. Они учитывают нюансы, не отраженные в данных (бюджетные ограничения, надежность поставщиков).

5. Принятие решения. На основе формального анализа и экспертных оценок ЛПР выбирает оптимальную стратегию.

### Обсуждение преимуществ

Предложенная архитектура обладает рядом ключевых преимуществ:

- целостность и надежность данных, т. е. обеспечивается реляционной СУБД PostgreSQL;

- гибкость и мощь аналитики, т. е. Python позволяет реализовать сложные алгоритмы анализа данных и интегрировать библиотеки ИИ;

- управление сложными задачами, т. е. концепция ЧМКИ позволяет распределять работу между экспертами и программными агентами, адаптируя процесс под требования конкретной проблемы;

- масштабируемость, т. е. система может масштабироваться как за счет увеличения мощности сервера БД, так и за счет добавления новых программных сервисов и привлечения большего числа экспертов.

### Заключение (Conclusion)

В данной статье была представлена комплексная разработка – интеллектуальная система поддержки принятия решений для управления машинными носителями информации. Исследование продемонстрировало эффективность интеграции трех ключевых технологий: реляционной базы данных PostgreSQL, языка программирования Python и концепции человеко-машинного коллективного интеллекта.

Разработанная реляционная схема базы данных обеспечивает надежное, структурированное и целостное хранение всей информации об МНИ, пользователях и их действиях. Реализованное на Python с использованием библиотеки rusorg2 приложение гарантирует безопасное, эффективное и отказоустойчивое взаимодействие с СУБД.

Главным научным и практическим вкладом работы является внедрение концепции ЧМКИ в предметную область управления МНИ. Это позволяет перейти от простой автоматизации учета к созданию адаптивной интеллектуальной среды, способной решать сложные, слабоформализованные задачи. Предложенная архитектура поддерживает самоорганизацию коллектива, объединяющего людей-экспертов и программные сервисы, что открывает новые возможности для повышения безопасности, оптимизации затрат и операционной эффективности.

Перспективы дальнейших исследований видятся в интеграции более сложных алгоритмов машинного обучения для прогнозирования инцидентов, развития онтологических моделей для улучшения семантической интероперабельности, а также в создании продвинутых инструментов визуализации для лиц, принимающих решения.

### Благодарности (Acknowledgement)

Коллектив авторов исследовательской работы выражает благодарность руководству ГБОУ АО ВО «АГАСУ» за финансовую, информационную и техническую поддержку, оказанную в ходе написания статьи.

### Список литературы

1. Бизнес-анализ деятельности интегрированных структур рыбохозяйственного комплекса России в условиях цифровизации / Н. А. Дубинина, О. Ю. Мичурина, О. В. Кудрявцева [и др.] // Инженерно-строительный вестник Прикаспия. – 2023. – № 2 (44). – С. 108–115. – DOI 10.52684/2312-3702-2023-44-2-108-115. – ЭДН ТППВТВ.
2. Retelny D. No workflow can ever be enough: How crowdsourcing workflows constrain complex work / D. Retelny, M. S. Bernstein, M. A. Valentine // *Proceedings of the ACM on Human-Computer Interaction*. – 2017. – Vol. 1, № 2.
3. Смирнов А. В. Поддержка принятия решений на основе человеко-машинного коллективного интеллекта: современное состояние и концептуальная модель / А. В. Смирнов, Т. В. Левашова, А. В. Пономарев // Информационно-управляющие системы. – 2020. – № 2. – С. 60–70.
4. *Mastering Data-Intensive Collaboration and Decision Making* / ed. N. Karacapilidis. – Springer, 2014. – 226 p.
5. Бялецкая Е. М. Оптимизация работы ТЭЦ на основании полученных данных из автоматизированной системы коммерческого учета электроэнергии (АСКУЭ) / Е. М. Бялецкая, Е. М. Дербасова // Инженерно-строительный вестник Прикаспия. – 2020. – № 4 (34). – С. 18–23.
6. Valentine M. A. Flash organizations / M. A. Valentine, D. Retelny, A. To, N. Rahmati, T. Doshi, M. S. Bernstein // *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. – 2017. – P. 3523–3537.
7. Буняева Е. В. Различия диалектов языка SQL в СУБД Microsoft SQL Server и Postgree SQL / Е. В. Буняева, А. Г. Романова // Развитие науки и практики в глобально меняющемся мире в условиях рисков (шифр – МКРНП) : сборник материалов XXIX Международной научно-практической конференции. – Москва, 2024. – С. 232–241.
8. Dignum F. Interactions as social practices: towards a formalization. – arXiv preprint arXiv:1809.08751, 2018.
9. Дзялошинский И. М. Когнитивные процессы человека и искусственный интеллект в контексте цифровой цивилизации : монография / И. М. Дзялошинский. – Москва : Ай Пи Ар Медиа, 2022. – 583 с. – ISBN 978-5-4497-1596-8. – EDN ESIFKW.
10. Дронов В. Ю. Python как средство автоматизации в информационной безопасности. защита доступа к базам данных в Python / В. Ю. Дронов, Г. А. Дронова // Динамика систем, механизмов и машин. – 2021. – Т. 9, № 4. – С. 54–59.
11. Кудрявцева О. В. Разработка базы данных в MongoDB для управления изменениями в DevOps / О. В. Кудрявцева, М. А. Кудрявцева, Е. М. Бялецкая // Инженерно-строительный вестник Прикаспия. – 2025. – № 2 (52). – С. 81–86. – DOI 10.52684/2312-3702-2025-52-2-81-86. – EDN CXPCBI.
12. Умарова У. М. Автоматизация процесса создания отчета / У. М. Умарова, В. А. Шапов, А. В. Тарутин // Автоматизированные системы управления и информационные технологии : материалы Всероссийской научно-технической конференции : в 2 т. – Пермь, 2020. – С. 316–319.
13. Ковтуненко А. А. Разработка DLP-модуля для защиты персональных данных в распределенной автоматизированной системе / А. А. Ковтуненко // Интеллектуальные технологии на транспорте. – 2024. – № 3 (39). – С. 65–72. – DOI 10.20295/2413-2527-2024-339-65-72. – EDN LRICVY.
14. Мересий А. В. Автоматизация тестирования систем управления облачными базами данных на примере DBaaSPostgresPro / А. В. Мересий, И. С. Полевщиков // Инженерный вестник Дона. – 2024. – № 8 (116). – С. 133–153. – EDN FYXPUP.
15. Еремеев Д. Е. Интеграция машинного обучения, баз данных и мониторинга проектов: создание эффективной системы управления данными / Д. Е. Еремеев // Молодые ученые в решении актуальных проблем науки : сборник материалов Всероссийской научно-практической конференции студентов, аспирантов и молодых ученых (с международным участием), Красноярск, 18–19 апреля 2024 года. – Красноярск : Сибирский государственный университет науки и технологий им. акад. М. Ф. Решетнева, 2024. – С. 580–581. – EDN GRFRHN.
16. Гибадуллин Р. Ф. Реализация механизма репликации в СУБД PostgreSQL / Р. Ф. Гибадуллин, А. М. Зиннатова, М. Ю. Перухин, Р. Н. Гайнуллин // Вестник Технологического университета. – 2020. – Т. 20, № 24. – С. 100–101.
17. Буняева Е. В. Различия диалектов языка SQL в СУБД Microsoft SQL Server и PostgreSQL / Е. В. Буняева, А. Г. Романова // Развитие науки и практики в глобально меняющемся мире в условиях рисков (МКРНП). – Москва, 2024. – С. 232–241.
18. Шуршев В. Ф. Оценка и управление рисками банкротства / В. Ф. Шуршев, О. В. Кудрявцева, И. И. Шукуров // Инженерно-строительный вестник Прикаспия. – 2022. – № 3 (41). – С. 109–113. – DOI 10.52684/2312-3702-2022-41-3-109-113. – EDN YKXGMI.
19. Алмазбек М. Принципы работы с библиотеками Python в прикладных исследованиях / М. Алмазбек, А. Д. Абалиева, Д. Р. Шамырова // Современные проблемы механики. – 2021. – № 43 (1). – С. 120–131. – EDN XLWIET..
20. Vocharov N. M. Python as a tool for creating programs / N. M. Vocharov // Студенческая наука - взгляд в будущее : материалы XVIII Всероссийской студенческой научной конференции, Красноярск, 15–17 марта 2023 года. – Красноярск : Красноярский государственный аграрный университет, 2023. – Часть 1. – С. 301–304. – EDN ALTFLG.

### References

1. Dubinina N. A., Michurina O. Yu., Kudryavtseva O. V., et al. Biznes-analiz deyatel'nosti integrirovannikh struktur ribokhozyaystvennogo kompleksa Rossii v usloviyakh tsifrovizatsii [Business analysis of the activities of integrated structures of the Russian fisheries complex in the context of digitalization]. *Inzhenerno-stroitel'nii vestnik Prikaspiya* [Engineering and Construction Bulletin of the Caspian Sea]. 2023, no. 2 (44), pp. 108–115. DOI 10.52684/2312-3702-2023-44-2-108-115. EDN TPPVTV.
2. Retelny D., Bernstein M. S., Valentine M. A. No workflow can ever be enough: How crowdsourcing workflows constrain complex work. *Proceedings of the ACM on Human-Computer Interaction*. 2017, vol. 1, no. 2.
3. Smirnov A. V., Levashova T. V., Ponomarev A. V. Decision Support Based on Human-Machine Collective Intelligence: Current State and Conceptual Model. *Information and Control Systems*. 2020, no. 2, pp. 60–70.

4. Karacapilidis N. (ed.) *Mastering Data-Intensive Collaboration and Decision Making*. Springer; 2014, 226 p.
5. Byaletskaya Ye. M., Derbasova Ye. M. Optimizatsiya raboti TETs na osnovanii poluchennikh dannikh iz avtomatizirovannoi sistemi kom-mercheskogo ucheta elektroenergii (ASKUE) [Optimization of the Thermal Power Plant Operation Based on the Data Obtained from the Automated Commercial Electricity Accounting System (ACUES)]. *Inzhenerno-stroitel'nyi vestnik Prikaspiya* [Engineering and Construction Bulletin of the Caspian Sea]. 2020, no. 4 (34), pp. 18–23.
6. Valentine M. A., Retelny D., To A., Rahmati N., Doshi T., Bernstein M. S. Flash organizations. *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 2017, pp. 3523–3537.
7. Bunyaeva Ye. V., Romanova A. G. Razlichiya dialektov yazika SQL v SUBD Microsoft SQL Server i Postgree SQL [Differences of dialects of the SQL language in the Microsoft SQL server and postgree SQL database systems]. *Razvitie nauki i praktiki v globalno menyayushchemsya mire v usloviyakh riskov (shifr – MKRNP) : sbornik materialov XXIX Mezhdunarodnoi nauchno-prakticheskoi konferentsii* [Development of Science and Practice in a Globally Changing World under Conditions of Risks (Code – MKRNP) : Collection of Materials of the XXIX International Scientific and Practical Conference]. Moscow; 2024, pp. 232–241.
8. Dignum F. *Interactions as social practices: towards a formalization*. arXiv preprint arXiv:1809.08751; 2018.
9. Dzyaloshinskii I. M. *Kognitivnie protsessy cheloveka i iskusstvennii intellekt v kontekste tsifrovoi tsivilizatsii* [Human Cognitive Processes and Artificial Intelligence in the Context of Digital Civilization]. Moscow: IPR Media; 2022, 583 p. ISBN 978-4497-1596-8. – EDN ESIFKW.
10. Dronov V. Yu., Dronova G. A. Python kak sredstvo avtomatizatsii v informatsionnoi bezopasnosti. zashchita dostupa k bazam dannikh v Python [Python as a means of automation in information security. protecting access to databases in Python]. *Dinamika sistem, mekhanizmov i mashin* [Dynamics of Systems, Mechanisms, and Machines]. 2021, vol. 9, no. 4, pp. 54–59.
11. Kudryavtseva O. V., Kudryavtseva M. A., Byaletskaya Ye. M. Razrabotka bazi dannikh v MongoDB dlya upravleniya izmeneniyami v DevOps [Development of a Database in MongoDB for DevOps Change Management]. *Inzhenerno-stroitel'nyi vestnik Prikaspiya* [Engineering and Construction Bulletin of the Caspian Sea]. 2025, no. 2(52), pp. 81–86. DOI 10.52684/2312-3702-2025-52-2-81-86. EDN CXPCBI.
12. Umarova U. M., Shchapov V. A., Tarutin A. V. Avtomatizatsiya protsessa sozdaniya otcheta [Automation of the report creation process]. *Avtomatizirovannye sistemi upravleniya i informatsionnye tekhnologii : materialy Vserossiiskoi nauchno-tekhnicheskoi konferentsii* [Automated Control Systems and Information Technologies : Materials of the All-Russian Scientific and Technical Conference]. Perm; 2020, pp. 316–319.
13. Kovtunenka A. A. Razrabotka DLP-modulya dlya zashchity personalnykh dannikh v raspredelennoi avtomatizirovannoi sisteme [Development of a DLP Module for Protecting Personal Data in a Distributed Automated System]. *Intellektualnye tekhnologii na transporte* [Intelligent Technologies in Transport]. 2024, no. 3 (39), pp. 65–72. DOI 10.20295/2413-2527-2024-339-65-72. EDN LRICVY.
14. Meresiy A. V., Polevshchikov I. S. Avtomatizatsiya testirovaniya sistem upravleniya oblachnimi bazami dannikh na primere DBaaSPostgresPro [Automation of Testing Cloud Database Management Systems on the Example of DBaaSPostgresPro]. *Inzhenernii vestnik Dona* [Engineering Bulletin of the Don]. 2024, no. 8(116), pp. 133-153. EDN FYXPUP.
15. Yermeev D. Ye. Integratsiya mashinnogo obucheniya, baz dannikh i monitoringa proektov: sozdanie effektivnoi si-stemi upravleniya dannimi [Integration of Machine Learning, Databases, and Project Monitoring: Creating an Effective Data Management System]. *Molodie uchenie v reshenii aktualnykh problem nauki : sbornik materialov Vserossiiskoi nauchno-prakticheskoi konferentsii studentov, aspirantov i molodikh uchenikh (s mezhdunarodnim uchastiem), Krasnoyarsk, 18–19 aprelya 2024 goda* [Young Scientists in Solving Current Scientific Problems : Collection of Materials from the All-Russian Scientific and Practical Conference of Students, Postgraduate Students, and Young Scientists (with International Participation), Krasnoyarsk, April 18–19, 2024]. Krasnoyarsk: Sibirskii gosudarstvennii universitet nauki i tekhnologii im. akad. M. F. Reshetneva; 2024, pp. 580–581. EDN GRFRHN.
16. Gibadullin R. F., Zinnatov A. M., Perukhin M. Yu. Realizatsiya mekhanizma replikatsii v SUBD PostgreSQL [Implementation of the replication mechanism in the PostgreSQL DBMS]. *Vestnik tekhnologicheskogo universiteta* [Bulletin of the Technological University]. 2020, vol. 20, no. 24, pp. 100–101.
17. Bunyaeva Ye. V., Romanova A. G. Razlichiya dialektov yazika SQL v SUBD Microsoft SQL Server i PostgreSQL [Differences in SQL dialects in Microsoft SQL Server and PostgreSQL DBMS]. *Razvitie nauki i praktiki v globalno menyayushchemsya mire v usloviyakh riskov (MKRNP)* [Development of Science and Practice in a Globally Changing World under Conditions of Risks (MKRNP)]. Moscow; 2024, pp. 232–241.
18. Shurshev V. F., Kudryavtseva O. V., Shukurov. I. I. Otsenka i upravlenie riskami bankrotstva [Assessment and management of bankruptcy risks]. *Inzhenerno-stroitel'nyi vestnik Prikaspiya* [Engineering and Construction Bulletin of the Caspian Sea]. 2022, no. 3 (41), pp. 109–113. DOI 10.52684/2312-3702-2022-41-3-109-113. EDN YKXGMI.
19. Almazbek M., Abaliev A. D., Shamurova D. R. *Printsipy raboti s bibliotekami Python v prikladnykh issledovaniyakh* [Principles of Working with Python Libraries in Applied Research]. *Sovremennye problemi mekhaniki* [Modern Problems of Mechanics]. 2021, no. 43 (1), pp. 120–131. EDN XLWIET.
20. Bocharov N. M. Python as a tool for creating programs. *Student Science – a Look into the Future : Materials of the XVIII All-Russian Student Scientific Conference, Krasnoyarsk, March 15–17, 2023*. Krasnoyarsk: Krasnoyarskii gosudarstvennii agrarnii universitet; 2023, part 1, pp. 301–304. EDN ALTFLG.

© О. В. Кудрявцева, Е.М. Бялецкая, М.А. Кудрявцева

**Ссылка для цитирования:**

Кудрявцева О. В., Бялецкая Е. М., Кудрявцева М. А. Интеллектуальная система поддержки принятия решений для управления машинными носителями информации на основе человеко-машинного коллективного интеллекта // Инженерно-строительный вестник Прикаспия : научно-технический журнал / Астраханский государственный архитектурно-строительный университет. Астрахань : ГБОУ АО ВО «АГАСУ», 2026. № 1 (55). С. 104–111.