

Научная статья | Original paper

УДК 004.652

Разработка интеллектуального ассистента для применения стандартизированных распределённых бизнес-моделей

И.Н. Фомин

Институт проблем точной механики и управления
Федерального исследовательского центра «Саратовский научный центр
Российской академии наук», Саратов, Российская Федерация
✉ ignik16@yandex.ru

Резюме

Контекст и актуальность. В условиях цифровой трансформации глобальной экономики ключевую роль играют индустриальные цифровые платформы и экосистемы, основанные на стандартизированных распределённых бизнес-моделях. Теоретической основой исследования послужила референтная модель RAMI 4.0, позволяющая интегрировать производственные активы в многомерную структуру. **Цель.** Разработать ИИ-ассистента для автоматизации ввода данных в референтную модель RAMI 4.0 и актуализации баз знаний в распределённых индустриальных экосистемах. **Гипотеза.** Интеллектуальная классификация элементов, определения координат в RAMI-кубе и валидация на основе онтологических моделей автоматизируют формирование баз знаний о производственных активах промышленных предприятий и повысит эффективность управления географически распределёнными производственными процессами. **Методы и материалы.** Произведён теоретико-информационный анализ концептов стандартизированной модели RAMI 4.0, в соответствии с ГОСТ Р 59799 и другими стандартами моделирования знаний, формализация функций ИИ-ассистента с использованием массивов данных и баз знаний о производственных активах, осуществлено моделирование процессов информационного обмена между субъектами экосистем индустриальных цифровых платформ. Реализован прототип интеллектуального ассистента на платформе Google Colab с библиотеками Python (Scikit-learn, NLTK, OWL API). Для обработки текстовых и лингвистических данных о производственных активах применены методы нечёткой логики. **Результаты.** Разработана онтологическая модель референтной модели RAMI 4.0, впервые определены её концепты, предикаты и их подклассы; формализованы функции ИИ-ассистента



с операциями над данными; предложена структура данных интеллектуального ассистента для его использования совместно с ERP-системами. **Выводы.** ИИ-ассистент обеспечивает автоматизированное наполнение баз знаний, способствуя формированию экосистем производственных предприятий и консорциумов, работающих с применением технологий «Индустрии 4.0».

Ключевые слова: ИИ-ассистент, RAMI 4.0, распределённые бизнес-модели, онтологическая модель, киберфизические системы, цифровые платформы

Для цитирования: Фомин, И.Н. (2026). Разработка интеллектуального ассистента для применения стандартизированных распределённых бизнес-моделей. *Моделирование и анализ данных*, 16(3), 121—154. <https://doi.org/10.17759/mda.2026160306>

Development of an intelligent assistant for applying standardized distributed business models

I.N. Fomin

Institute of Precision Mechanics and Control Problems

Saratov Scientific Center of the Russian Academy of Sciences, Russian Federation

✉ ignik16@yandex.ru

Abstract

Context and relevance. In the context of the digital transformation of the global economy, industrial digital platforms and ecosystems based on standardized distributed business models play a key role. The theoretical foundation of the study is the Reference Architectural Model Industrie 4.0 (RAMI 4.0), which enables the integration of production assets into a multidimensional structure. **Objective.** To develop an AI assistant for automating data entry into the RAMI 4.0 reference model and updating knowledge bases in distributed ecosystems. **Hypothesis.** Intelligent classification of elements, determination of coordinates in the RAMI cube, and validation based on ontological models will automate the creation of knowledge bases about production assets and enhance the efficiency of cyber-physical systems management. **Methods and materials.** Theoretical-information analysis of the concepts of the standardized RAMI 4.0 model in accordance with IEC 62890 and other knowledge modeling standards; formalization of the AI assistant functions using data arrays and knowledge bases on production assets; modeling of information exchange processes between participants in industrial digital platform ecosystems. The prototype was implemented on the Google Colab platform using Python libraries (Scikit-learn, NLTK, OWL API). Fuzzy logic methods were applied to process textual and linguistic data about production assets. **Results.** An ontological model of the RAMI 4.0 reference model has been developed, with its concepts, predicates,



and subclasses defined for the first time; the functions of the AI assistant have been formalized along with operations on data; a data structure has been proposed for integrating the AI assistant with ERP systems. **Conclusions.** The AI assistant enables automated knowledge base population, contributing to the formation of ecosystems of production enterprises and consortia operating with Industry 4.0 technologies.

Keywords: AI assistant, RAMI 4.0, distributed business models, ontological model, cyber-physical systems, digital platforms

For citation: Fomin, I.N. (2026). Development of an intelligent assistant for applying standardized distributed business models. *Modelling and Data Analysis*, 16(3), 121–154. (In Russ.). <https://doi.org/10.17759/mda.2026160306>

Введение

Глобальная экономика переживает значительные изменения, переходя от модели, основанной на эффекте масштаба, к системам, где ключевую роль играют малые агенты (Müller, 2019). В условиях формирования постиндустриального общества возникают новые бизнес-модели, в которых роботы и интеллектуальные ассистенты выполняют важнейшие этапы бизнес-процессов, что изменяет курс научно-технического развития предприятий, отраслей, государства¹. Цифровая трансформация открывает перспективы для внедрения методов искусственного интеллекта (ИИ) и больших языковых моделей (LLM-моделей), которые становятся важным инструментом для повышения скорости и эффективности экономических процессов.

Применение этих технологий не только обеспечивает эффективное взаимодействие и координацию внутри предприятий и консорциумов, между различными подразделениями компаний, расположенными в разных регионах и государствах, но и способствует формированию индустриальных экосистем (Pauli et al., 2021). Такие экосистемы представляют собой взаимосвязанные сети, в которых участники, включая поставщиков, клиентов и контрагентов, могут взаимодействовать друг с другом, обмениваться данными и ресурсами, совместно разрабатывать новые продукты и услуги.

Таким образом, киберфизические системы и цифровые платформы становятся основными строительными блоками, на которых базируются инновационные бизнес-модели, способствующие устойчивому развитию и конкурентоспособности географически распределённых предприятий в условиях быстро меняющегося рынка.

Индустриальная цифровая платформа — это цифровая экосистема, в которой производители оборудования, разработчики, поставщики услуг и клиенты совместно создают и используют ценность на основе общих стандартов и данных (Müller, 2019). Цифровая экосистема для индустриальных цифровых платформ — это совокупность независимых акторов (производителей оборудования, разработчиков программного

¹ Правительство РФ, О Стратегии развития информационного общества в Российской Федерации на 2017–2030 годы: Указ Президента РФ № 203 от 09.05.2017. Собрание законодательства Российской Федерации от 15.05.2017. № 20. 2901 с.



обеспечения, поставщиков услуг, интеграторов и промышленных потребителей), которые используют общую платформу для совместной разработки и использования решений на основе промышленных данных (Pauli et al., 2021).

Для разработки цифровых платформ, обеспечивающих оптимизацию производственной деятельности и формирование экосистем взаимодействия, ведущие технологические компании (Cisco, General Electric, IBM, Google, Яндекс) объединяют усилия в рамках специализированных консорциумов.

На самом высоком этапе развития платформы начинают формировать межотраслевые экосистемы, известные как «экострии» или «эгополии» (рис. 1).

Развить экосистему до экострии или эгополии можно, применяя технологии промышленного интернета вещей (IIoT), при условии, что участники цифровых платформ действуют в рамках эталонной стандартизированной распределённой бизнес-модели.

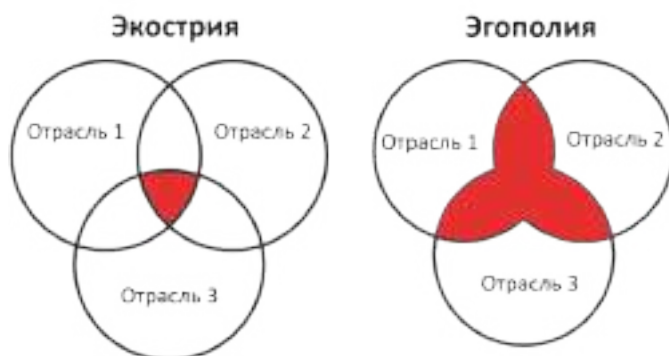


Рис. 1. Различия между высшими стадиями развития экосистем, формирующихся цифровыми бизнес-моделями посредством глобальных цифровых платформ

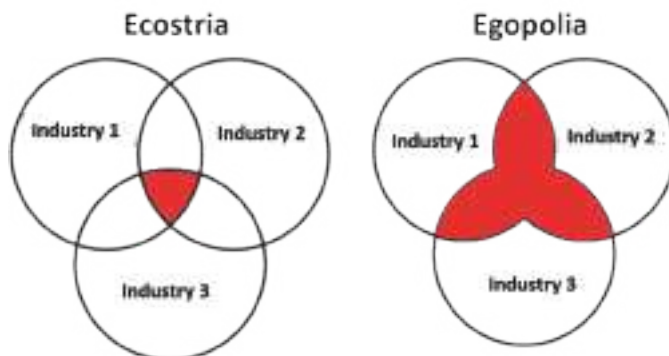


Fig. 1. Differences between the higher stages of development of ecosystems formed by digital business models through global digital platforms



Эталонная (референтная) стандартизированная распределённая бизнес-модель — это обобщённая, типовая модель, описывающая роли участников индустриальной цифровой экосистемы при создании совместного продукта, а также принципы управления и монетизации, используемая как методологический ориентир при проектировании и производстве сложных технических систем. В контексте применения цифровых платформ и экосистем такая бизнес-модель является универсальной, абстрактной схемой, описывающей типовые роли участников, их ценностные предложения, способы создания и распределения ценности (Каленов, 2022). Такого рода экосистемно-ориентированные бизнес-модели строятся вокруг логики совместного создания и распространения ценности между участниками (Rong et al., 2023).

На текущем этапе развития технологий интеллектуального производства существует несколько ведущих стандартизированных референтных моделей, ориентированных на распределённые бизнес-модели с применением технологий ПоТ и «Индустрию 4.0». Эти модели используются различными высокотехнологичными консорциумами. Как показано в работе (Фомин, 2025), наиболее востребованной, структурно проработанной и подходящей для реализации в России является концепция референтной модели Representation of Reference Architecture Industry 4.0 (RAMI 4.0).

RAMI описывает производственные активы через три ключевых аспекта, образующих так называемый многомерный RAMI-куб: измерения, определяющие уровни (слои), производственных активов, оборудования, систем и бизнес-процессов (Hierarchy Levels); измерения, определяющие отношение актива к организационной структуре предприятий, находящихся в общей экосистеме (Integration Layers); измерения, определяющие жизненный цикл, продукции и элементов систем (Life Cycle).

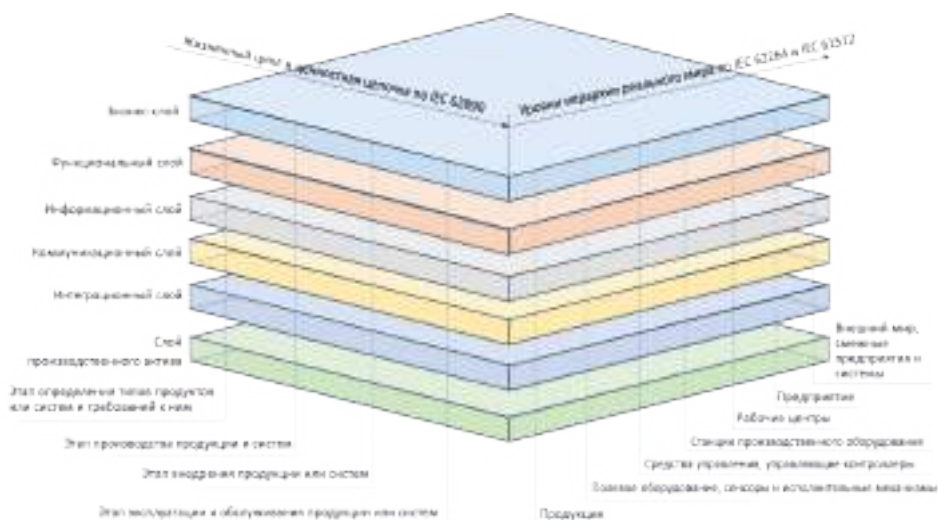


Рис. 2. Представление эталонной архитектуры «Индустрия 4.0» по стандарту IEC 63088, в виде RAMI-куба

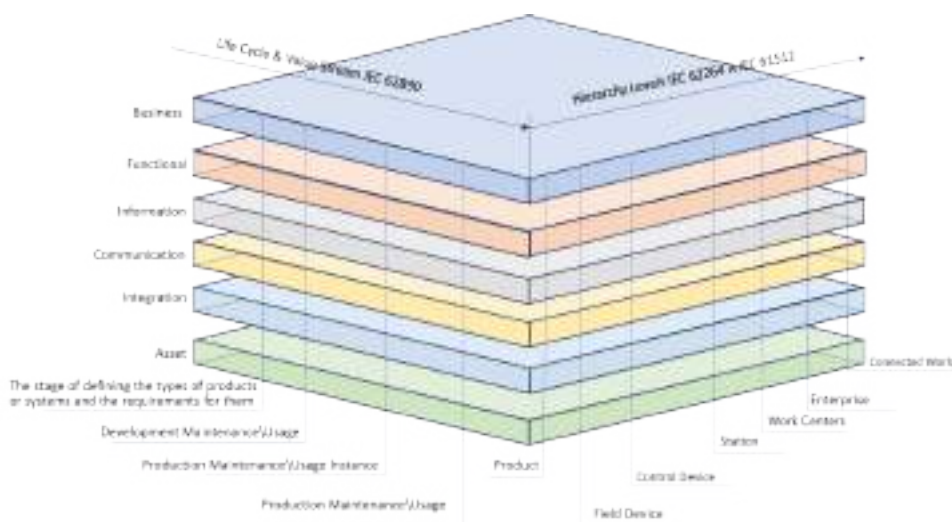


Fig. 2. Representation of the “Industry 4.0” Reference Architecture according to IEC 63088 as the RAMI Cube

Такой подход позволяет интегрировать любой актив или технический компонент с другими смежными активами и математически описать координаты любого элемента в RAMI-кубе (рис. 2). Определить эти координаты для сотен тысяч элементов производственных активов — очень сложная задача. Она требует формальную классификацию элементов, определение координат в многомерной модели, валидацию на соответствия стандартам и техническим требованиям. Ранее эти функции успешно автоматизировались применением методов анализа данных для формирования онтологии предметной области (Антонов, Воронов, 2011). На современном этапе всё чаще подобные функции возлагаются на интеллектуального ассистента (ИИ-ассистента), который может автоматически описывать элементы активов и технических систем, расписывая их по измерениям RAMI-куба.

В отличие от предыдущих исследований, фокусирующихся на теоретических аспектах RAMI (Megow, 2020), настоящая работа предлагает практическую реализацию ИИ-ассистента для автоматизации процессов. Главная гипотеза: автоматизация на основе онтологических моделей повысит эффективность. Цели: формализовать функции ИИ-ассистента, разработать модель данных и прототип.

Функции ИИ-ассистента для ввода данных в референтную модель RAMI 4.0

AI-ассистент — это программное обеспечение, которое самостоятельно выполняет набор различных функций посредством управления различными



специализированными программами или контроллерами промышленных цифровых платформ, обладающее возможностью обучаться на накопленных данных или, используя предобученные модели, принимать решения по своим действиям на основе собранной информации и управлять бизнес-процессами. AI- или ИИ-ассистентов уже начали применять в логистике, бухгалтерском учёте, в энергосберегающем инжиниринге, в управлении качеством. ИИ-ассистент может автоматически находить объекты, неотражённые в референтной модели цифровой платформы, и вносить их в соответствующие базы данных, наполняя базу знаний о том или ином объекте. Для исследовательских задач такого рода ассистента можно реализовать на общедоступной платформе Google Colab.

ИИ-ассистент, предназначенный для актуализации баз знаний, используя базы знаний, основанные на онтологических моделях, должен выполнять следующие ключевые функции:

1. **Настройка ИТ-среды и загрузка сведений о модели RAMI:** инициализирует программную среду, устанавливает необходимые фреймворки и библиотеки, формирует прикладную модель RAMI 4.0.
2. **Предобработка текстовых описаний и интерпретация нечётких требований:** выполняет предварительную обработку текстового описания нового элемента (например, «интеллектуальный переключатель») и описаний категорий многомерной модели RAMI (например, «эксплуатация, использование, работа»).
3. **Классификация элемента:** определяет тип элемента (например цифровой или физический актив предприятия).
4. **Определение координат в многомерной модели:** устанавливает координаты элемента по трём измерениям, в соответствии с концепцией RAMI 4.0.
5. **Интеграция элемента в многомерную модель:** добавляет новый элемент как экземпляр нового актива (например, «интеллектуальный переключатель» как экземпляр физических активов предприятия), определяет связи и предикаты новых элементов с другими элементами систем.
6. **Валидация соответствия стандартам:** проверяет корректность интеграции элемента, сравнивая координаты и связи с требованиями стандартов (ГОСТ Р 59799, IEC 62890, IEC 62264, IEC 61512 и др.), и вносит корректировки при необходимости.
7. **Представление результатов пользователю** в виде таблиц или дашбордов.
8. **Корректировка на основе обратной связи:** принимает уточнённое описание элемента от пользователя (например, «интеллектуальный переключатель в разработке»), повторно выполняет классификацию и интеграцию, обновляя многомерную модель при необходимости.

Чтобы определиться с методами и инструментами разработки AI-ассистента с такими функциями требуется выполнить формальную постановку задачи, в которой произвести отнесение вышеперечисленных функций AI-ассистента к его функциональным блокам и к массивам данных, которые они должны обрабатывать.



Для формализации технической задачи разработки AI-ассистента определим массивы данных, которые должны храниться и обрабатываться AI-ассистентом для выполнения своих функций. Для этого определим необходимые данные для каждой функции AI-ассистента.

Для настройки программной среды ИИ-ассистента потребуются набор данных о производственных активах, которые описаны референтной моделью, её концептов, в соответствии с требованиями стандартов. Это массив данных D_1 . Главной функцией AI-ассистента является наполнение или обновление сформированной модели в массиве D_2 и формирование набора связей, которые могут быть описаны на естественном языке, в виде тезауруса D_3 . Эти связи и отношения должны формировать векторные представления терминов из тезауруса в виде массива данных об их векторных представлениях $D_{3,1}$.

Пользователь должен вводить текстовое описание активов (элементов и объектов систем), формируя массив данных D_4 , приводя его к тезаурусу D_3 , а AI-ассистент должен формализовать исходные описания категорий, в соответствии с установками, заданными в массиве D_1 , формируя массив данных D_5 с исходным описанием различных параметров, включая тип, класс и категорию элементов, которые вошли в D_1 . Обработка данных D_3 , D_4 , D_5 и формирует массив векторных представлений $D_{3,1}$.

На основании определённых координат и связей элементов, их стадий жизненного цикла Life Cycle, уровня иерархии Hierarchy Levels и слоёв архитектуры Integration Layers, должен формироваться общий массив данных, в который входят классы, связи элементов и их координаты по осям RAMI-куба. Именно эта информация должна визуализироваться для пользователя, валидироваться, корректироваться и обновлять онтологию D_2 .

Для решения этой задачи применяются современные модели эмбедингов, которые способны учитывать семантическую близость терминов и контекст употребления слов. Это позволяет повысить качество поиска и классификации элементов по сравнению с традиционными статистическими методами. Для этого применяют модели Sentence-BERT (Reimers, Gurevych, 2019), E5 9. (Wang, Yang, 2022) и BGE (Xiao, Liu, 2024), которые демонстрируют высокую эффективность в задачах семантического поиска, информационного извлечения и сопоставления текстовых описаний объектов. Применение классических операций обработки текстов с созданием TF-IDF-векторов для создания прототипов ИИ-ассистентов тоже вполне оправдано (Поляков, 2025).

Проведя теоретико-информационный анализ референтной модели, связывающей элементы систем в терминах многомерной RAMI-модели на первом этапе исследований были формализованы функции и связанные с ними массивы данных ИИ-ассистента (таб. 1). Таблица показывает идентифицированные виды данных, которые формируются и обрабатываются на разных этапах функционирования интеллектуального ассистента.

Такое представление данных дало возможность операции первичной настройки ИИ-ассистента для формирования массивов D_3 , D_4 , D_5 обозначить через a_1 , а операцию обработки текстов с созданием TF-IDF-векторов — через a_2 . При этом



разрабатываемой информационной системе потребуется использование операции объединения a_7 для добавления элементов в онтологию, операции проекции a_3 — a_6 для извлечения координат, операций фильтрации a_8 для проверки, коррекции и преобразования информации, а также операций a_9 , которая преобразует данные в визуализированную форму для представления пользователям. В случае, если пользователь будет производить уточнённое описание элементов a_9 , то потребуется использование массива данных с уточнённым описанием D_8 , которые обновляют ранее созданный тезаурус D_3 . Результаты проведённого теоретико-информационного анализа процессов и функций ИИ-ассистента представлены в таб. 1.

Для реализации интеллектуального ассистента в виде информационной системы формализуем вышеописанные процессы обработки данных.

Для реализации гибкой многомерной RAMI-модели для изменяющихся и управляемых элементов систем определим совокупность взаимосвязанных областей знаний или семантических пространств, которые описывают различные аспекты производственных активов в рамках референтной архитектуры RAMI 4.0 в виде множества доменов, которые можно реализовать на платформе Google Colab.

Включим в эти домены сущности, определяемые российскими и международными стандартами²:

- классы активов (например, физический или цифровой актив), уровни иерархии (уровень устройств, уровень оборудования или уровень предприятия);
- слои интеграции (это активы, или их процессы коммуникации, или бизнес-процессы предприятия и т.п.);
- этапы жизненного цикла (разработка, эксплуатация, техническое обслуживание и т. п.)³;
- динамические характеристики (например, изменяющиеся состояния надежности или производительности).

Применение этих сущностей в онтологической модели ИИ-ассистента позволит каждое множество доменов определить через базу знаний. В такой базе знаний домены могут быть связаны предикатами, включать разношкальные переменные (числовые и лингвистические) с функциями принадлежности для учета неопределенности параметров активов и связей (Дли М.И., 2023), а также применять так называемые эволюционные и квазигенетические алгоритмы, обеспечивающие генерацию последовательности запросов, организованных по специальной логической схеме, включающие, с последующим анализом подобранных ИИ-ассистентом данных методами многомерного статистического анализа (Куравский, Л.С. и др. 2025).

² DIN SPEC 91345:2016–04. Referenzarchitekturmodell Industrie 4.0 (RAMI 4.0).

³ Росстандарт. ГОСТ Р 59799–2021. Умное производство. Модель эталонной архитектуры индустрии 4.0 (RAMI 4.0).



Таблица 1

Функции и связанные с ними массивы данных AI-ассистента

Виды массивов данных	Примеры данных	Настройка AI-ассистента									
		Создание TF-IDF — векторов	Классификация элементов	Координаты Life Cycle	Координаты Hierarchy Levels	Определение координат по Integration Layers	Интеграция в онтологическую модель	Валидация и проверка	Представление результатов	Уточнение модели	
		a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}
Данные общей мета-онтологии	D_1	Описание категорий (например, «FieldDevice: интеллектуальный переключатель»)									
	D_2	Общая онтология RAMI 4.0 (цифровая модель актива)									
	D_3	Словарь, термины, синонимы, отношения									
	$D_{3,1}$	Вектора элементов и категорий									
	D_4	Текстовое описание элемента (например, «интеллектуальный переключатель»)									
	D_5	Категории по исходной модели									
	D_6	Новые категории (например, Operation, FieldDevice, Asset)									
	D_7	Численные характеристики элемента (например, параметры переключателя или датчика)									
Уточнённое описание элементов	D_8	Внутренние модели данных (например, характеристики цифрового двойника)									



Table 1

Functions and associated data arrays of the AI assistant

Types of Data Arrays		Examples of Data	AI Assistant Setup										Integration into the Ontological Model	Validation and Verification	Presentation of Results	Model Refinement
			D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8	D_9	D_{10}				
General Meta-Ontology Data	D_1	Description of categories (for example, "FieldDevice: intelligent switch")														
Ontology	D_2	General RAMI 4.0 Ontology (digital asset model)														
Thesaurus	D_3	Vocabulary, terms, synonyms, relationships														
Elements and Categories	$D_{3,1}$	Vectors of Elements and Categories														
Textual Description of Elements	D_4	Textual description of an element (for example, "intelligent switch")														
Initial Description of Categories	D_5	Categories According to the Original Model														
Coordinates Along the Axes of the RAMI Cube	D_6	New categories (for example, Operation, FieldDevice, Asset)														
Ontology Visualization	D_7	Numerical characteristics of an element (for example, switch or sensor parameters)														
Refined Description of Elements	D_8	Internal Data Models (for example, characteristics of a digital twin)														



Формальная постановка задачи ввода информации об элементах технических систем в структуру цифровых платформ

Пусть U — универсальное множество, содержащее все возможные данные, с которыми может работать система (тексты, онтологии, векторы, координаты и т.д.). Опираясь на функции AI-ассистента, концепты и предикаты онтологической модели RAMI, выделим подмножества в виде массивов данных для каждого хранилища и внешней сущности:

- $D_1 \subseteq U$: — множество данных из «Общая мета-онтология, стандарты IEC 62890, IEC 62264, IEC 61512» (стандарты и концепты).
- $D_2 \subseteq U$: — множество данных «Обновлённая онтология» (инстанции, классы, предикаты).
- $D_3 \subseteq U$: — множество данных «Тезаурус» (лингвистические связи, синонимы).
- $D_{3.1} \subseteq U$: — множество данных «TF-IDF-векторы элементов и категорий» (векторные представления)
- $D_4 \subseteq U$: — множество данных «Текстовое описание элементов» (исходные тексты пользователей)
- $D_5 \subseteq U$: — множество данных «Исходное описание категорий» (описания классов из стандартов).
- $D_6 \subseteq UD$: — множество данных «Координаты и связи элементов по осям RAMI-куба» (классификации, координаты, связи).
- $D_7 \subseteq U$: — множество данных «Визуализированная представленная онтология» (результаты визуализации).
- $P \subseteq U$: — множество данных от внешней сущности «Пользователь» (ввод текста и обратная связь).
- $R \subseteq U$: — множество данных для внешней сущности «Пользователь» (результаты или вывод визуализированной онтологии).

Каждый процесс информационной системы a_i ($i = 1 \dots 10$) можно представить как отображение (функцию) между множествами: $a_i: A_i \rightarrow B_i$, где A_i — входные множества, B_i — выходные множества. Тогда общий процесс можно представить как композицию функций:

$$F = a_{10} \circ a_9 \circ a_8 \circ a_7 \circ a_6 \circ a_5 \circ a_4 \circ a_3 \circ a_2 \circ a_1,$$

где вход — $P \cup D_1 \cup D_8$, а выход — $R \cup D_7$. После представления результатов обработки данных предполагается возможность уточнения модели пользователем, т.е. предусмотрена итерация через a_{10} (таб. 4), которая добавляет обратную связь: $a_{10}(P) \rightarrow D_3 \rightarrow a_2$, образуя цикл.

Пусть $S = D_1 \cup D_2 \cup D_{3.1} \cup D_3 \cup D_4 \cup D_5 \cup D_{3.2} \cup D_6 \cup D_7$ — множество состояний системы, состоящее из разных массивов данных, которые необходимо обрабатывать AI-ассистенту. Тогда изменение состояний на этапах функционирования системы можно представить как переход состояний: $S_{t+1} = \text{Update}(S_t, a_i(A_i))$, где a_i — текущий процесс.



В таких обозначениях, формальной задачей становится разработать модель последовательности обработки информации AI-ассистентом, который осуществляет переходы состояний системы посредством осуществления операций над множествами (массивами данных), определёнными в таб. 1:

- объединение (используется в a_7 для добавления элементов в онтологию);
- проекция (используется в функциях a_3 — a_6 для извлечения координат);
- фильтрация (используется в функции a_8 для проверки и коррекции);
- преобразование (в функции a_2 преобразует текст в векторы, а в функции a_9 преобразует данные в визуализацию).

Структура онтологической модели цифровой платформы на основе референтной бизнес-модели

Как было сказано выше, промышленные цифровые платформы функционируют на основе общих стандартов и данных. Общие данные служат основой формирования баз знаний о производственных активах, оборудовании, технических системах, элементов продукции. Управление базами знаний производится с применением онтологического подхода.

Онтологический подход подразумевает структурирование и организацию знаний в виде формальных моделей, которые описывают понятия и их взаимосвязи в определенной области. В контексте географически распределённых производственных предприятий, использующих концепцию «Индустрии 4.0», онтологический подход позволяет создать единую модель, которая будет объединять данные о различных аспектах производственных процессов, оборудовании и продуктах.

При проектировании структуры данных киберфизической системы промышленных предприятий, находящихся в одной экосистеме, с применением онтологического подхода, требуется:

- определить концепты и понятия в терминологии RAMI4.0 (например, «машина», «процесс», «продукт») и их свойства, определить методы борьбы с неоднозначностью.
- определить связи между понятиями, которые могут быть заданы текстом и лингвистически разными постановщиками задач, инженерами, на разных предприятиях (при этом один объект может быть «частью» другого, или «использоваться» для выполнения определённого процесса).
- обеспечить унификацию и иерархичность различных сведений о реальных и абстрактных сущностях (активах), применяющихся в производственных процессах, являющихся элементами цифровых моделей предприятий, производственных процессов и производимых продуктов.
- обеспечить возможность расширения онтологий новыми понятиями и отношениями, что позволяет адаптироваться к меняющимся требованиям и новым тенденциям в области управления знаниями.



В результате применения онтологического подхода в контексте «Индустрии 4.0» предприятия могут более эффективно хранить, обрабатывать и обмениваться данными, тем самым повышая уровень автоматизации, улучшая принятие решений на основе данных и оптимизируя производственные процессы.

Онтологическая модель индустриальной платформы должна состоять из мета-онтологии, которую организует администратор цифровой платформы и прикладных эталонных моделей, которые формируются производственными предприятиями, входящими в экосистему цифровой платформы.

Для разработки модели данных ИИ-ассистента потребовалось формализовать классифицированный список концептов онтологической модели цифровой платформы на основе стандартизированной бизнес-модели RAMI 4.0 (табл. 2).

Таблица 2 / Table 2

Концепты онтологической модели RAMI 4.0
Concepts of the RAMI 4.0 ontological model

Концепт / The concept	Комментарий / Comment
Жизненный цикл и ценностная цепочка по IEC 62890 / The life cycle and value chain according to IEC 62890	
Типы /Type	Описывает тип информации или данных, используемых в системе / Describes the type of information or data used in the system
Примеры / Instance	Конкретные реализации или экземпляры типов / Specific implementations or instances of types
Иерархические уровни по IEC 62264 / Hierarchical levels according to IEC 62264	
Продукция (Product)	Описывает физические объекты, продукцию / Describes physical objects, products
Полевое оборудование / Field Device	Описывает элементы производственных линий, интеллектуальные переключатели и исполнительные механизмы / Describes the elements of production lines, intelligent switches, and actuators
Средства управления / Control Device)	Описывает управляющие контроллеры / Describes the control controllers
Производственные станции / Station	Описывает рабочие станции или станки или агрегаты / Describes workstations, machines, or units
Производственные линии и рабочие центры / Work Centers	
Предприятие / Enterprise	Описывает структуры и элементы, относящиеся к уровню управления предприятием / Describes the structures and elements related to the enterprise management level
Внешний мир / Connected World	Описывает сведения о смежных предприятиях и системах / Describes information about related enterprises and systems
Архитектура по EC 61512 / Architecture according to EC 61512	
Активы /Asset	Физические, абстрактные и цифровые активы / Physical, abstract, and digital assets



Концепт / The concept	Комментарий / Comment
Продукт / Product	Продукты и их компоненты / Products and their components
Интеграция / Integration	Интеграция данных и систем / Integration of data and systems
Коммуникация / Communication	Обмен данными и информацией / Data and information exchange
Информация / Information	Управление данными и информацией / Data and information management
Функциональная безопасность / Functional Safety	Обеспечение безопасности систем / Ensuring system security
Бизнес / Business	Управление бизнес-процессами / Business process management
Уровни интеграции по IEC 62890 / Integration levels according to IEC 62890	
Слой производственного актива / Digital Representation	Описывает цифровые модели и двойники / Describes digital models and digital twins.
Интеграционный слой / Integration	Описывает элементы, обеспечивающие интеграцию разных систем и данных / Describes the elements that ensure the integration of different systems and data
Коммуникационный слой / Communication	Описывает физические и логические соединения / Describes physical and logical connections
Информационный слой / Information	Данные, информация, массивы данных / Data, information, arrays of data
Функциональный слой / Functional	Описывает функции и сервисы элементов систем / Describes the functions and services of system elements
Бизнес-слой / Business	Описывает бизнес-процессы и модели / Describes business processes and models

Выявленные свойства позволили определить оптимальный набор предикатов, которые могут автоматически сформировать цифровую модель элемента производственной системы в виде онтологии. При разработке онтологической модели вышеопределённые концепты были связаны между собой в онтологическую модель через набор предикатов. При этом, каждому из предполагаемых предикатов определялся квантор существования и квантор общности, а затем предикаты были распределены по категориям. Это дало возможность не только упростить их использование, но и реализовать структуру данных в ERP-системе, в соответствии с разработанной системой предикатов.

Эксперименты с онтологическими моделями, построенными с помощью больших языковых моделей популярных нейросетей и редакторами онтологий позволили впервые сформировать структуру предикатов, которая, вместе с структурой классов полностью формирует онтологическую модель, на которой может быть построена структура знаний об элементах технических систем, объединённых в промышленную цифровую платформу (таб. 3).



Таблица 3

Предикаты онтологической модели RAMI 4.0

Класс	Подкласс	Предикат
Соответствует (correspondsTo)	Соответствует этапам жизненного цикла (IEC 62890)	Соотносит тип актива с этапом жизненного цикла (isTypeOf)
		Определяет регламентирующий стандарт для этапа жизненного цикла элемента системы (conformsTo)
		Определяет этап жизненного цикла (hasLifeCycleStage)
	Соответствует уровням иерархии (IEC 62264 и IEC 61512)	Соотносит тип актива с уровнем иерархии (isTypeOf)
		Определяет уровень иерархии (belongsToLevel)
Взаимодействует (interactsWith)	Взаимодействует на этапах жизненного цикла (IEC 62890)	Определяет элементы, с которым взаимодействует актив на разных этапах ЖЦ (interactsWith)
		Определяет элементы, с которыми актив взаимодействует на физическом уровне (hasPhysicalConnection)
		Определяет элементы, с которыми актив взаимодействует на цифровом уровне (hasDigitalConnection)
		Определяет требования к синхронизации физического и цифрового актива (synchronizesWith)
	Взаимодействует с объектами (IEC 62264 и IEC 61512)	Определяет компоненты или системы, которые выдают управляющие воздействия (controlledBy)
		Определяет компоненты или системы, которые собирают данные об исполнении функций (controls)
		Определяет этапы процесса, на которых происходит взаимодействие с активом (executesProcess)
Управляет (controls)	Управляет этапами жизненного цикла (IEC 62890)	Определяет фазы жизненного цикла, на которых происходит управление активом (executesProcess)
		Определяет требования к управляющим воздействиям (controlledBy)
		Определяет требования к информации об исполнении функций (controls)
	Управляет объектами (IEC 62264 и IEC 61512)	Определяет функции управления активом (executesProcess)
		Определяет управляющий элемент для актива (controlledBy)
		Определяет управляемый элемент для актива (controls)



Table 3

Predicates of the RAMI 4.0 ontological model

Class	Subclass	Predicate
Corresponds (correspondsTo)	Corresponds to life cycle stages (IEC 62890)	Relates the asset type to a life cycle stage (isTypeOf)
		Defines the governing standard for the system element's life cycle stage (conformsTo)
		Defines the life cycle stage (hasLifeCycleStage)
	Corresponds to hierarchy levels (IEC 62264 and IEC 61512)	Relates the asset type to a hierarchy level (isTypeOf)
		Defines the hierarchy level (belongsToLevel)
Interacts (interactsWith)	Interacts at life cycle stages (IEC 62890)	Defines the elements with which the asset interacts at different life cycle stages (interactsWith)
		Defines the elements with which the asset interacts at the physical level (hasPhysicalConnection)
		Defines the elements with which the asset interacts at the digital level (hasDigitalConnection)
		Defines the synchronization requirements for the physical and digital asset (synchronizesWith)
	Interacts with objects (IEC 62264 and IEC 61512)	Defines the components or systems that issue control actions (controlledBy)
		Defines the components or systems that collect data on function execution (controls)
		Defines the process stages at which interaction with the asset occurs (executesProcess)
Controls (controls)	Controls life cycle stages (IEC 62890)	Defines the life cycle phases at which the asset is controlled (executesProcess)
		Defines the requirements for control actions (controlledBy)
		Defines the requirements for information about function execution (controls)
	Controls objects (IEC 62264 and IEC 61512)	Defines the asset control functions (executesProcess)
		Defines the controlling element for the asset (controlledBy)
		Defines the controlled element for the asset (controls)



Несколько десятков предикатов невозможно отразить в рамках статьи, поэтому в таб. 4 приведён укрупнённый классификатор классов предикатов. Каждый подкласс предикатов в этой таблице может содержать от двух до пяти предикатов, которые с высокой точностью определяют координаты любого элемента технической системы в многомерном RAMI-кубе. Например, подкласс «Передаёт данные на этапы ЖЦ», в соответствии с ГОСТ Р 59799, IEC 62264 и IEC 61512, содержит набор предикатов «Определяет требования к указанию объектов и компонентов, в адрес которых актив передаёт информацию (sendsDataTo)», «Определяет требования актива передавать сведения в смежные системы и компоненты (transfersDataTo)», «Определяет требования к хранению данных на разных этапах жизненного цикла (storesDataIn)».

Таблица 4

Подклассы предикатов онтологической модели RAMI 4.0

Класс	Подкласс
Соответствует (correspondsTo)	Соответствует этапам жизненного цикла (IEC 62890)
	Соответствует уровням иерархии (IEC 62264 и IEC 61512)
Взаимодействует (interactsWith)	Взаимодействует на этапах жизненного цикла (IEC 62890)
	Взаимодействует с объектами (IEC 62264 и IEC 61512)
Управляет (controls)	Управляет этапами жизненного цикла (IEC 62890)
	Управляет объектами (IEC 62264 и IEC 61512)
Поддерживает (supports)	Поддерживает этапы жизненного цикла (IEC 62890)
	Поддерживает функционирование (IEC 62264 и IEC 61512)
Связывает (links)	Связывает с этапами жизненного цикла (IEC 62890)
	Соответствует с уровнями иерархии (IEC 62264 и IEC 61512)
Оптимизирует (optimizes)	Оптимизирует этапы жизненного цикла (IEC 62890)
	Оптимизирует по уровням иерархии (IEC 62264 и IEC 61512)
Получает данные (receivesDataFrom)	Получает данные об этапах ЖЦ (IEC 62890)
	Получает данные от объектов на уровнях иерархии (IEC 62264)
Передаёт данные (transfersDataTo)	Передаёт данные на этапы ЖЦ (IEC 62264 и IEC 61512)
	Передаёт данные в объекты на уровнях иерархии (IEC 62264)
Описывает (describes)	Описывает этапы жизненного цикла (IEC 62890)
	Описывает уровни иерархии (IEC 62264 и IEC 61512)
Запускает (triggers)	Запускает на этапах жизненного цикла (IEC 62890)
	Запускает на уровнях иерархии (IEC 62264 и IEC 61512)
Предоставляет возможность (enables)	Предоставляет возможность на этапах ЖЦ (IEC 62890)
	Предоставляет возможность на уровнях иерархии (IEC 62264)



Table 4

Subclasses of predicates of the RAMI 4.0 ontological model

Class	Subclass
Corresponds (correspondsTo)	Corresponds to life cycle stages (IEC 62890)
	Corresponds to hierarchy levels (IEC 62264 and IEC 61512)
Interacts (interactsWith)	Interacts at life cycle stages (IEC 62890)
	Interacts with objects (IEC 62264 and IEC 61512)
Controls (controls)	Controls life cycle stages (IEC 62890)
	Controls objects (IEC 62264 and IEC 61512)
Supports (supports)	Supports life cycle stages (IEC 62890)
	Supports functioning (IEC 62264 and IEC 61512)
Links (links)	Links with life cycle stages (IEC 62890)
	Corresponds to hierarchy levels (IEC 62264 and IEC 61512)
Optimizes (optimizes)	Optimizes life cycle stages (IEC 62890)
	Optimizes by hierarchy levels (IEC 62264 and IEC 61512)
Receives Data (receivesDataFrom)	Receives data about life cycle stages (IEC 62890)
	Receives data from objects at hierarchy levels (IEC 62264)
Transfers Data (transfersDataTo)	Transfers data to life cycle stages (IEC 62264 and IEC 61512)
	Transfers data to objects at hierarchy levels (IEC 62264)
Describes (describes)	Describes life cycle stages (IEC 62890)
	Describes hierarchy levels (IEC 62264 and IEC 61512)
Triggers (triggers)	Triggers at life cycle stages (IEC 62890)
	Triggers at hierarchy levels (IEC 62264 and IEC 61512)
Enables (enables)	Enables at life cycle stages (IEC 62890)
	Enables at hierarchy levels (IEC 62264)

Чтобы обеспечить актуальность онтологической модели и учёт лингвистически заданных разношкальных переменных, нужно обеспечить множество доменов возможностью их изменения с учетом динамических изменений состояний активов через временные метки с применением правил нечеткой логики. Например, домен «FieldDevice» может включать подмножества интеллектуальных переключателей и исполнительных механизмов, каждый из которых имеет свои атрибуты (напряжение, сила тока, температура, надежность) и связи с другими доменами (например, с «ControlDevice» через предикат «sendsDataTo»). Подход, при котором каждый домен служит источником прецедентной информации для анализа и принятия решений, позволяет структурировать данные в онтологии и поддерживать так называемые алгоритмы оценки близости прецедентов (Дли М.И., 2023).

При такой организации данных заказчик или производитель изделий, находящейся в экосистеме индустриальной цифровой платформы, может определить основные сведения о требуемой или производимой продукции, задав её лингвистическое описание, жизненный цикл, элементы изделия и связанные с ними элементы, согласно п. 4.1 стандартов IEC 63088 и ГОСТ 59799, включая средства и методы производства.



Такой набор концептов в сложной многомерной иерархии, элементы которого неоднозначно связаны друг с другом предикатами, относящимися к более чем к десяти классам, иллюстрирует сложность цифровых моделей элементов технических систем и показывает на сколько трудозатратно присвоить новому элементу киберфизической системы соответствующие координаты RAMI-куба. Задача усложняется тем, что эти элементы являются элементами базы данных различных ERP-систем предприятий, входящих в экосистему, могут быть заданы различными людьми в текстовом и лингвистическом формате.

На современном этапе развития больших языковых моделей (LLM) задачи наполнения баз знаний об элементах технических систем, образующих киберфизические системы под управлением промышленных цифровых платформ, могут решаться применением так называемых AI-ассистентов.

Разработка модели данных для реализации AI-ассистента

Для разработки модели данных и реализации функций AI-ассистента требуется связать ранее определённые функциональные блоки потоками данных, определив входные и выходные массивы данных для каждого блока и операции с массивами данных, осуществляемые на каждом этапе обработки информации, в соответствии с таб. 4.

При осуществлении функции a_1 «Настройка AI-ассистента», её вход можно определить как $A_1 = P \cup D_1$ (текст от пользователя и данные из мета-онтологии или стандартов), а выход как $B_1 = D_2 \cup D_5 \cup D_3$ (инициализация онтологии, категорий, тезауруса и текста). Если представить эту функцию, как $a_1: P \times D_1 \rightarrow D_2 \times D_5 \times D_3$ то операции проекции и объединения в ней покажут массивы данных и их значения:

$$a_1(p, d_1) = (d_2^0, d_5^0, d_3^0),$$

где $d_4 = p$ (прямое копирование текста), а d_{2a}^0, d_5^0, d_3^0 — начальные значения из D_1 .

При осуществлении функции a_2 «Создание TF-IDF-векторов», её вход: $A_2 = D_4 \cup D_5 \cup D_3$ представляется в виде текста с описанием элементов, с отнесением к соответствующей категории в тезаурусе. Её выходы $B_2 = D_{3.1}$ — это векторные представления элементов из тезауруса. Если представить её функцию как $a_2: D_4 \times D_5 \times D_3 \rightarrow D_{3.1}$, то операция интерпретации нечётких требований и преобразования текста в векторы с учётом тезауруса, в этой функции:

$$a_2(d_4, d_5, d_3) = d_{3.1},$$

где $d_{3.1}$ — множество TF-IDF-векторов, полученных с использованием тезауруса для обогащения текста.

При осуществлении функции a_3 «Классификация элементов» мы имеем вход: $A_3 = D_{3.1}$ в виде векторных представлений элементов из тезауруса и выход: $B_3 = D_6$, формирующий начальную классификацию этих элементов. Эту функцию можно представить



как $a_3: D_{3,1} \rightarrow D_6$ с операцией классификации $a_3(d_{3,1}) = d_6^0$, где d_6^0 — множество пар (элемент, класс), определённых через сходство векторов (например, косинусное расстояние).

При осуществлении функций a_5 «Определение координат по Hierarchy Levels», a_6 «Определение координат по Integration Layers» и a_4 «Определение координат по Life Cycle» их входами должны стать множества TF-IDF-векторов, полученных с использованием тезауруса для обогащения текста $d_{3,1}$:

$$\begin{aligned} a_4: D_{3,1} \rightarrow D_6: a_4(d_{3,1}) &= \{(e, c, (lc, \emptyset, \emptyset), \emptyset) \mid (e, c, \emptyset, \emptyset) \in d_6, lc = \arg \max_{lc} \text{score}(v_e, lc')\}; \\ a_5: D_{3,1} \rightarrow D_6: a_5(d_{3,1}) &= \{(e, c, (lc, hl, \emptyset), \emptyset) \mid (e, c, \emptyset, \emptyset) \in d_6, hl = \arg \max_{hl} \text{score}(v_e, hl')\}; \\ a_6: D_{3,1} \rightarrow D_6: a_6(d_{3,1}) &= \{(e, c, (lc, hl, il), \emptyset) \mid (e, c, \emptyset, \emptyset) \in d_6, il = \arg \max_{il} \text{score}(v_e, il')\}, \end{aligned}$$

где score — функция сходства (например, косинусное расстояние), а выходами d_6 становятся координаты и связи элементов по трём осям RAMI-куба:

$$d_6^1, d_6^2, d_6^3 = \left\{ \begin{aligned} &d_6^1 \cup \{(e, cl) \mid e \in d_6^1, cl \in \text{LifeCycle}(d_{3,1})\} \\ &d_6^2 \cup \{(e, hl) \mid e \in d_6^2, hl \in \text{HierarchyLevels}(d_{3,1})\} \\ &d_6^3 \cup \{(e, il) \mid e \in d_6^3, il \in \text{IntegrationLayers}(d_{3,1})\} \end{aligned} \right\}$$

В этом выражении e — это конкретный элемент или актив в системе, который представлен в множестве D_6 . Это может быть, например, физический объект (интеллектуальный переключатель, или другое устройство) или цифровая сущность (программное обеспечение), описываемый в рамках цифровой платформы или киберфизической системы. В контексте RAMI — e может быть идентификатором актива (например, «Sensor1» или «Controller2»), который проходит классификацию и привязку к координатам в трехмерной модели (жизненный цикл lc , уровень иерархии hl , слой интеграции il).

При осуществлении функции функций a_7 «Интеграция в онтологическую модель» её входами становятся координаты и классификация элементов из массива D_6 , лингвистические отношения и семантические связи элементов из D_3 , а также элементы всей онтологической модели из D_2 , т.е. $A_7 = D_6 \cup D_2 \cup D_3$. При этом, выходами этой функции становятся те же массивы данных $B_7 = D_2 \cup D_6$ в обновлённом виде. Таким образом, эту функцию можно представить как $a_7: D_6 \times D_2 \times D_3 \rightarrow D_2 \times D_6$, в которой должны осуществляться операции объединения и преобразования: $a_7(d_6^3, d_2, d_3) = (d_2', d_6^4)$, где $d_2' = d_2 \cup \text{Integrate}(d_6^3, d_3)$, а d_6^4 — обновлённые данные с дополнительными связями.

После обновления данных в массивах D_2 и D_6 требуется произвести валидацию и проверку вновь внесённых данных на непротиворечивость данных. При осуществлении функции a_8 (валидация и проверка) её вход можно интерпретировать как $A_8 = D_6 \cup D_2$ (координаты и онтология), а выход как: $B_8 = D_6 \cup D_2$ (проверенные данные, опционально исправленная онтология). Тогда функция: $a_8: D_6 \times D_2 \rightarrow D_6 \times D_2$ осуществляет операцию фильтрации и коррекции: $a_8(d_6^4, d_2') = (d_6^5, d_2'')$, где d_6^5 — проверенные данные, а $d_2'' = d_2'$ или $d_2' \cup \text{Correct}(d_6^4, d_2')$ при необходимости.

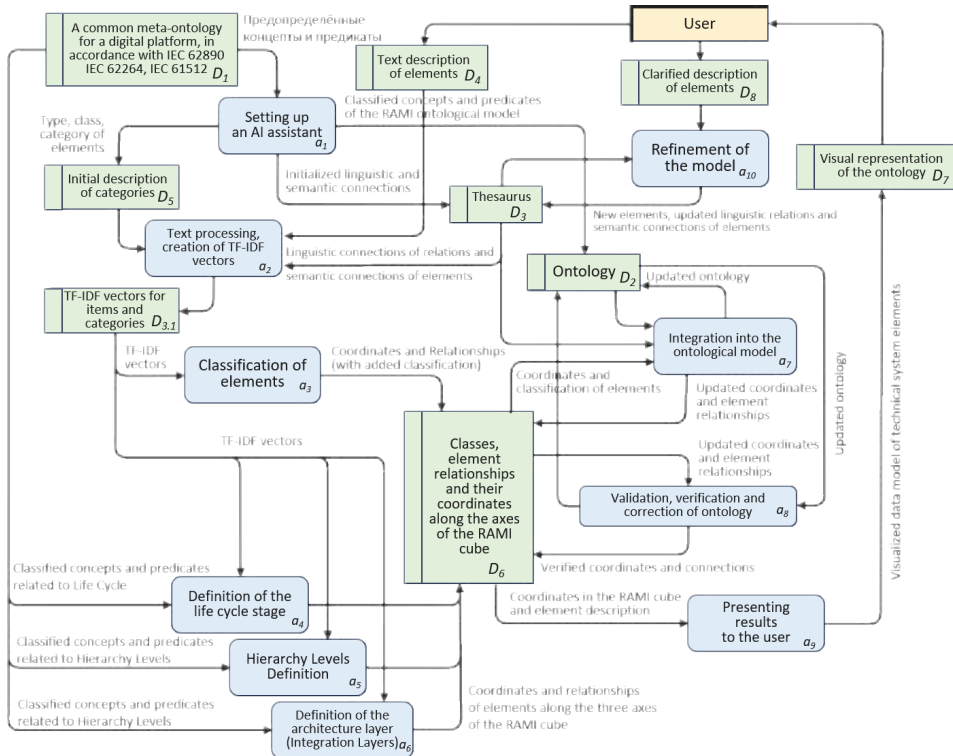


Fig. 3. Data structure of the AI assistant for intelligent control of the RAMI 4.0 reference model in DFD notation

В разработанной модели для осуществления функций a_8 и a_{10} ИИ-ассистент может применять различные методы оптимизации, в зависимости от инструментов, на которых он реализуется. Самый подходящий метод, при этом, является итеративный метод решения задач и вычислений, который заключается в последовательном применении одной или нескольких операций (шагов) к начальному значению, чтобы постепенно приближаться к желаемому результату. Этот метод можно применить, используя стандартные библиотеки Python в платформе Google Colab. Применив этот метод оптимизации в функциях a_8 и a_{10} , на каждом шаге результат предыдущей итерации будет использоваться как входное значение для следующей.

Итеративный процесс помогает определить фиксированные точки с устойчивым состоянием системы, что важно в моделировании динамических систем, в управлении киберфизическими системами по RAMI 4.0, где состояния активов эволюционируют.

Если $S = D_1 \cup D_2 \cup D_{3.1} \cup D_3 \cup D_4 \cup D_5 \cup D_{3.1} \cup D_6 \cup D_7$ — множество состояний системы, состоящее из разных массивов данных, которые необходимо обрабатывать ИИ-ассистенту, то итерация S_{t+1} формирует композицию функций F посредством итерации:



$$S_{t+1} = F(S_t, a_{10}(P)).$$

При этом функция a_{10} должна обновить массив данных D_3 , и процесс должен возвратиться к функции обработки текстов для создания TF-IDF-векторов a_2 .

Информационная система, осуществляющая добавление новых активов и объектов и обновление связей достигает фиксированной точки, если $S_{t+1} = S_t$. В этом случае D_3 перестаёт обновляться (обратная связь завершена). Т.е., формально:

$$S^* = \text{fix}(F), \text{ где } \text{fix}(F) = \{S \mid F(S) = S\}.$$

Если пользователь вносит конечное множество уточнений P в функции a_{10} и, при этом, инициализированные лингвистические и семантические связи, созданные функцией a_1 тоже конечно, то процесс сходится за конечное число итераций. Метрика сходимости при этом $\text{dist}(S_t, S_{t+1}) = |D_3^{t+1} \setminus D_3^t|$, а сходимость достигается, когда $\text{dist}(S_t, S_{t+1}) = 0$, что фиксируется функцией a_8 .

При разработке структуры данных для реализации ИИ-ассистента были определены и апробированы инварианты для проверки корректности:

— Инвариант 1- Сохранение исходного текста:

$$\forall t: D_4^t = D_4^0, \text{ где } D_4^0 \text{ — начальное множество текстов.}$$

— Инвариант 2 — Соответствие координат и онтологии:

$$\forall (e, c, (lc, hl, il), r) \in D_6^t: \exists (e, \text{hasLifeCycleStage}, lc) \in D_2^t$$

— Инвариант 3 — Монотонное обновление тезауруса:

$$\forall t: D_3^t \subseteq D_3^{t+1}.$$

При апробации, инвариант 1 обеспечивал гарантию целостности, проверяя, что исходные данные не искажаются. Инвариант 2 контролировал согласованность структуры RAMI 4.0 и онтологии D_2 . Инвариант 3 гарантировал монотонность развития тезауруса D_3 , контролируя, что развитие ведётся только путем добавления новых связей, что важно для стабильности модели. Применённая в функции a_8 операция IsCorrect призвана выполнять проверку инвариантов 1, 2 и 3. Она была реализована в ИИ-ассистенте как часть итеративного процесса валидации и корректировки, чтобы обеспечить корректность онтологической модели на каждой итерации. В таб. 5 собраны методы и инструменты, которыми были реализованы различные функции ИИ-ассистента.

Таблица 5

Методы и инструменты, которыми были реализованы функции AI-ассистента

Функции AI-ассистента		Выполняемые действия	Методы и инструменты для реализации функций
a_1	Настройка AI-ассистента	Инициализация среды, загрузка мета-онтологии	Python, OWL API, Protégé, Google Colab
a_2	Создание TF-IDF-векторов	Преобразование текста в векторные представления	Scikit-learn (TF-IDF), NLTK, Python



Функции AI-ассистента		Выполняемые действия	Методы и инструменты для реализации функций
a_3	Классификация элементов	Определение типа элемента (PhysicalAsset и т.д.)	Косинусное сходство, кластеризация (K-means), Python
a_4	Определение координат по Life Cycle	Привязка к этапам жизненного цикла	Логика разметки OWL, семантический анализ
a_5	Определение координат по Hierarchy Levels	Привязка к уровням иерархии	Логика разметки OWL, RDF-графы
a_6	Определение координат по Integration Layers	Привязка к слоям интеграции	Логика разметки OWL, семантический анализ
a_7	Интеграция в онтологическую модель	Добавление элемента и предикатов в онтологию	OWL API, SPARQL, RDF-обновление
a_8	Валидация и проверка	Проверка корректности и инвариантов	Итеративная валидация, логические правила (SWRL)
a_9	Представление результатов	Визуализация онтологии для пользователя	Graphviz, D3.js, Google Colab (визуализация)
a_{10}	Уточнение модели	Обновление тезауруса по обратной связи	NLP (NLTK), итеративное обучение, Python

Table 5

Methods and tools used to implement the AI assistant's functions

AI Assistant Functions		Performed Actions	Methods and Tools for Implementing Functions
a_1	AI Assistant Setup	Environment initialization, meta-ontology loading	Python, OWL API, Protégé, Google Colab
a_2	Creation of TF-IDF Vectors	Converting text into vector representations	Scikit-learn (TF-IDF), NLTK, Python
a_3	Classification of Elements	Determining the element type (PhysicalAsset, etc.)	Cosine similarity, clustering (K-means), Python
a_4	Determining Coordinates for Life Cycle	Mapping to life cycle stages	OWL annotation logic, semantic analysis
a_5	Determining Coordinates for Hierarchy Levels	Mapping to hierarchy levels	OWL annotation logic, RDF graphs
a_6	Determining Coordinates for Integration Layers	Mapping to integration layers	OWL annotation logic, semantic analysis
a_7	Integration into the Ontological Model	Adding the element and predicates to the ontology	OWL API, SPARQL, RDF update
a_8	Validation and Verification	Checking correctness and invariants	Iterative validation, logical rules (SWRL)
a_9	Presentation of Results	Ontology visualization for the user	Graphviz, D3.js, Google Colab (visualization)
a_{10}	Model Refinement	Updating the thesaurus based on feedback	NLP (NLTK), iterative learning, Python



Результаты исследований и разработки ИИ-ассистента

Проведённые исследования структуры знаний, соответствующих требованиям RAMI позволили создать прототип ИИ-ассистента и апробировать модель двустороннего интеллектуального информационного обмена между корпоративными ERP-системами участников индустриальной экосистемы и центральной онтологией индустриальной цифровой платформы (ИЦП), построенной в соответствии с референтной архитектурной моделью RAMI 4.0.

Ключевым результатом исследования стало создание механизма, позволяющего автоматически формировать и актуализировать элементы баз знаний платформы на основе данных, поступающих из локальных ERP-систем, а также обратно — предлагать и создавать новые объекты в ERP на основе обновлений в онтологии ИЦП. Центральными компонентами системы обмена выступают:

- центральная информационная система ИЦП,
- корпоративная ERP-система участника,
- ИИ-ассистент, выполняющий семантическую обработку, классификацию и валидацию.

Модель бизнес-процесса информационного обмена представлена на рисунке 4 (для сценария создания нового элемента в ERP).

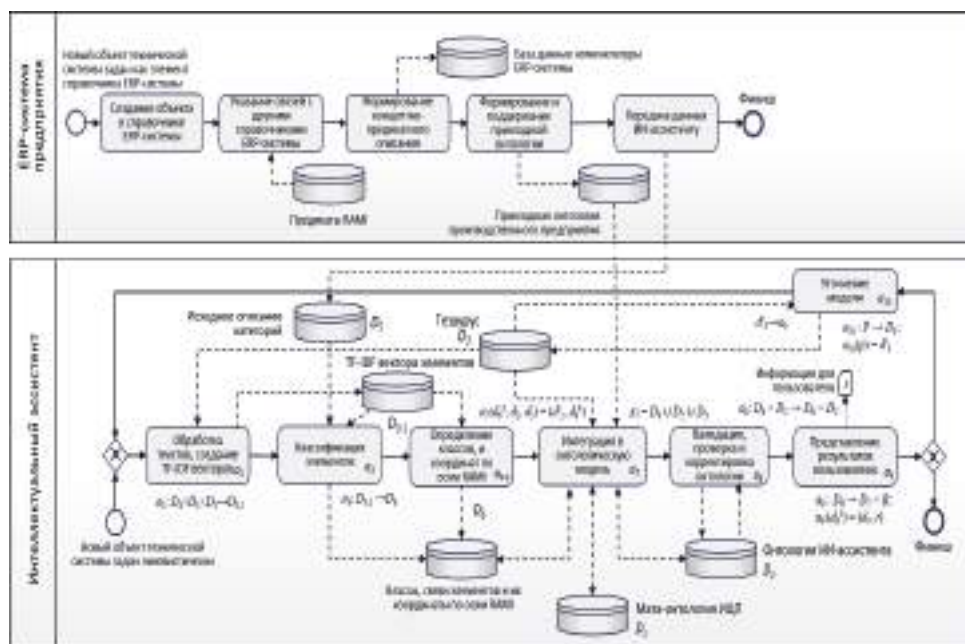


Рис. 4. Модель бизнес-процесса информационного обмена ИИ-ассистента и ERP-системы в нотации BPMN 2.0

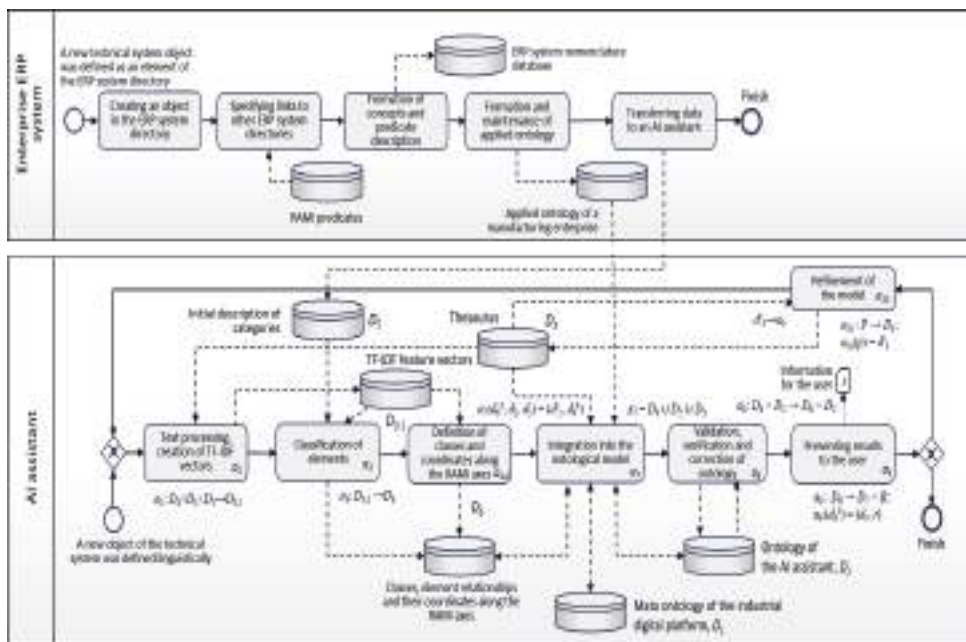


Fig. 4. Business process model of information exchange between an AI assistant and an ERP system in BPMN 2.0 notation

При создании нового элемента технической системы (оборудование, материал, узел и т.п.) в ERP-системе оператор выполняет стандартные действия:

- создаёт/выбирает объект в справочнике «Номенклатура» (или аналогичном),
- указывает связи с другими объектами через предикаты из специализированного справочника «Предикаты RAMI».

Если описание объекта задано на естественном языке (в поле технического описания или комментария), ИИ-ассистент автоматически выполняет полный цикл обработки, описанный в разделе «Функции ИИ-ассистента для ввода данных в референтную модель RAMI 4.0». Таким образом, обеспечивается автоматическое создание и интеграция нового элемента базы знаний платформы без ручного ввода координат и связей.

При появлении нового элемента в онтологии ИЦП (в результате действий других участников экосистемы или аналитиков платформы) автоматически инициируется:

- рассылка уведомлений участникам в форматах TXT или XML,
- интерпретация ИИ-ассистентом параметров элемента (координаты RAMI, связи, предикаты, атрибуты),
- проверка уникальности в локальной базе предприятия,
- автоматическое формирование предложения по созданию соответствующего объекта в ERP-системе.



Это даёт возможность оператору ERP-системы оценить готовое предложение с заполненными видом номенклатуры значениями по осям RAMI, набором предикатов для связей. После подтверждения или корректировки оператором предложенной информации, он запускает процедуру создания нового объекта в базе данных ERP, который может быть включён в состав цифровой модели изделия, предприятия или производственного процесса.

Для практической реализации двунаправленного обмена данными была разработана оригинальная конфигурация ERP-платформы «1С: Предприятие». Основными новациями в этой конфигурации стали:

- дополнительный справочник «Предикаты RAMI», который содержит предикаты онтологической модели с привязкой к осям RAMI 4.0 и реквизитами (тип связи, обязательность и др.).
- дополнительный справочник «Свойства RAMI» — позволяет оператору (или автоматически) указывать значения по трём осям модели при создании/редактировании объекта номенклатуры.

Интеграция структуры данных ERP-системы предприятия с базой знаний цифровой платформы осуществляется через внешние обработки или веб-сервисы. Выгрузка данных из 1С → передача ИИ-ассистенту → возврат обогащённых триплетов и координат RAMI → запись в дополнительные реквизиты объектов 1С. Обратный поток: приём уведомлений от цифровой платформы → интерпретация ИИ-ассистентом → предложение создания объекта в 1С.

Предложенный подход позволяет полностью автоматизировать извлечение концептов и связей из неструктурированных лингвистических описаний, присвоение координат RAMI без ручного вмешательства с низкой вероятностью ошибки, создание и синхронизацию элементов баз знаний как в локальных ERP, так и в центральной онтологии платформы.

Апробация на синтетических и частично реальных данных показала, что ИИ-ассистент способен автоматически формировать корректные элементы баз знаний в 75–85% случаев (при описаниях средней сложности), значительно сокращая время и трудозатраты на интеграцию данных в экосистему «Индустрии 4.0».

Обсуждение результатов

Главными результатами проведённого исследования и разработанного на его основе прототипа интеллектуального ассистента стало онтологическое описание референтной модели RAMI 4.0, в соответствии со стандартами ГОСТ Р 59799 и IEC 62890, которое позволяет применять положения этих стандартов с помощью популярной платформы «1С: Предприятие». Полученные результаты полностью подтверждают основную гипотезу исследования: интеллектуальная классификация элементов производственных активов, автоматическое определение их координат в многомерном RAMI-кубе и валидация на основе онтологических моделей позволяют автоматизировать создание и актуализацию баз знаний о киберфизических



системах в промышленных цифровых платформах, повышая эффективность управления распределёнными экосистемами «Индустрии 4.0». Дополнительные гипотезы также нашли подтверждение: использование TF-IDF-векторизации в сочетании с тезаурусом и элементами нечёткой логики обеспечивает надёжную обработку неструктурированных лингвистических описаний (точность автоматической классификации и привязки к осям RAMI достигла 75–85% на тестовых наборах), а двунаправленный обмен данными между ERP-системами и центральной онтологией платформы реализуем без нарушения целостности знаний.

Разработанный прототип ИИ-ассистента демонстрирует значительное сокращение трудозатрат: добавление нового элемента в базу знаний экосистемы (присвоение RAMI-координат), ранее требовавшее 12–25 минут ручной работы высоко квалифицированного аналитика, теперь занимает 18–45 секунд. Операция может выполняться в автоматическом режиме под контролем оператора низкой квалификации. Это согласуется с общим трендом цифровизации, где семантические технологии и онтологии выступают ключевым инструментом для преодоления гетерогенности данных в киберфизических системах.

Сходство проведённого исследования с результатами других исследований проявляется в использовании онтологического подхода для представления активов и процессов в рамках RAMI 4.0. В публикациях IEEE, в частности в (Awad 2016) подчёркивается роль формальных онтологий как семантической основы для обмена данными между интеллектуальными устройствами и системами, что полностью соответствует предложенному в настоящем исследовании решению с использованием мета- и прикладных онтологий.

Аналогично, в исследованиях (Bousdekis, 2021) и (Yahya, 2021) акцент делается на построении «графов знаний» на базе RAMI 4.0 для обеспечения интероперабельности и поддержки принятия решений в условиях больших данных. Предложенное техническое решение развивает эти идеи, добавляя автоматизированную обработку естественного языка и двунаправленную синхронизацию с корпоративными ERP-системами.

Кроме того, большинство существующих отечественных и зарубежных решений (например, OntoMat 4.0 или InPro ontology) фокусируются либо на статическом представлении знаний, либо на интеграции уже структурированных данных, не предлагая решений по автоматическому извлечению концептов из произвольных текстовых описаний объектов знаний. Предложенный в настоящем исследовании механизм предварительной обработки (лемматизация + TF-IDF + тезаурус) с последующим определением координат RAMI через векторное сходство и онтологический вывод позволяет обрабатывать реальные, часто неидеальные описания активов, вводимые персоналом без специальной подготовки. Это расширяет применимость подхода в реальных географически распределённых консорциумах, где данные поступают от разных предприятий в разнородных форматах.

Достигнутый уровень автоматизации, сокращение времени, повышение точности, открывают перспективы для дальнейшего развития: интеграция больших языковых



моделей (LLM) для улучшения интерпретации сложных технических текстов, добавление динамических аспектов (временные метки, нечёткие состояния активов) и масштабирование на межотраслевые экосистемы («экострии» и «экосиполии»).

Заключение

В результате проведённого исследования разработана онтологическая модель базы знаний для референтной архитектурной модели RAMI 4.0, формализованы ключевые функции ИИ-ассистента по автоматизированному вводу, классификации, определению координат в RAMI-кубе и валидации элементов производственных активов. Создана структура данных и реализован прототип ИИ-ассистента на платформе Google Colab, обеспечивающий двунаправленный интеллектуальный обмен между ERP-системой «1С: Предприятие» и центральной онтологией индустриальной цифровой платформы. Полученные результаты полностью подтверждают выдвинутую гипотезу: интеллектуальная обработка лингвистических описаний производственных активов, технических систем и их элементов с использованием TF-IDF-технологии, тезауруса и онтологического вывода позволяет автоматизировать создание и актуализацию баз знаний о киберфизических системах, существенно повышая эффективность управления распределёнными экосистемами «Индустрии 4.0».

Значимость работы заключается в преодолении барьеров гетерогенности данных и ручного труда при интеграции активов в многомерную модель RAMI 4.0, что особенно актуально для географически распределённых предприятий и консорциумов в условиях ускоренной цифровой трансформации. Предложенный подход решает поставленные цели: автоматизация ввода данных в индустриальные цифровые платформы и поддержка актуальности знаний в распределённых экосистемах.

Полученные результаты открывают перспективы для решения более сложных междисциплинарных задач: обеспечение семантической интероперабельности в глобальных цепочках поставок, поддержка автономных киберфизических систем с элементами самоорганизации, а также интеграция с цифровыми двойниками и генеративными ИИ-моделями для прогнозирования состояний активов.

Перспективы дальнейшей работы включают: интеграцию больших языковых моделей (LLM) для повышения точности интерпретации сложных технических текстов, внедрение динамических аспектов (временные метки, нечёткие состояния активов с использованием расширенной нечёткой логики), полномасштабное тестирование в реальных географически распределённых производственных предприятиях.

Ограничения. Исследование имеет ряд ограничений, которые следует учитывать при обобщении и практическом применении полученных результатов.

1) Отсутствие полномасштабного тестирования на реальных производственных данных.

Апробация прототипа ИИ-ассистента проводилась преимущественно на синтетических наборах данных (описания оборудования и материалов, сгенерированные на основе



типовых технических паспортов и стандартов) и ограниченном объёме реальных описаний (45 записей из открытых источников и партнёрских предприятий). Это не позволяет в полной мере оценить устойчивость алгоритмов к реальному разнообразию стилей описаний, профессиональному жаргону, ошибкам операторов и очень большим объёмам данных, характерным для действующих производственных консорциумов.

2) Отсутствие оценки в условиях реального времени и высокой нагрузки.

Прототип реализован на платформе Google Colab и не подвергался нагрузочному тестированию (одновременная обработка сотен/тысяч запросов от разных предприятий, обработка в реальном времени при частоте обновлений активов 1–10 в минуту).

3) Зависимость от качества тезауруса и мета-онтологии.

Точность автоматической интерпретации лингвистических описаний (75–85% в тестовых наборах) сильно зависит от полноты и актуальности унифицированного тезауруса и мета-онтологии RAMI 4.0. В реальных экосистемах с отраслевой спецификой (машиностроение, нефтехимия, энергетика и др.) потребуется постоянное расширение и поддержка этих словарей, что не было предметом настоящего исследования.

4) Отсутствие соответствия требованиям информационной безопасности.

Прототип без учёта требований к промышленной кибербезопасности (IEC 62443, ГОСТ Р 56939 и пр.). Обмен данными между ERP и ИЦП через внешние обработки и веб-сервисы потенциально создаёт уязвимости, которые требуют отдельного анализа и мер защиты.

5) Ограниченная поддержка динамических и нечётких состояний активов.

Хотя в прототипе использованы элементы нечёткой логики для обработки неопределённостей в описаниях, полноценный учёт динамических характеристик (изменение состояний активов во времени, деградация, переход между стадиями жизненного цикла) реализован лишь частично и требует дальнейшей разработки.

Указанные ограничения не уменьшают значимости полученных результатов как концептуальной и прототипной основы, но подчёркивают необходимость последующих этапов: полномасштабного пилотного внедрения в реальных производственных экосистемах, нагрузочного тестирования и проверки на безопасность, расширения на другие ERP-платформы и интеграции с современными большими языковыми моделями для повышения робастности обработки текстов.

Limitations. The study has several limitations that should be considered when generalizing and applying the results in practice.

1) Lack of full-scale testing on real production data.

The AI assistant prototype was validated mainly on synthetic datasets (equipment and material descriptions generated from typical technical passports and standards) and a limited set of real descriptions (45 records from open sources and partner enterprises). This does not allow a full assessment of algorithm robustness to real-world variability in description styles, domain-specific jargon, operator errors, and very large data volumes typical of operating production consortia.



2) No evaluation under real-time and high-load conditions.

The prototype was developed on Google Colab and has not undergone stress testing (simultaneous processing of hundreds/thousands of requests from different enterprises, real-time processing at asset update rates of 1–10 per minute). Performance, latency, and resource consumption in an industrial environment remain unknown.

3) Dependence on the quality of the thesaurus and meta-ontology.

The accuracy of linguistic description interpretation (75–85% in test sets) strongly depends on the completeness and currency of the unified thesaurus and RAMI 4.0 meta-ontology. In real ecosystems with industry-specific features (mechanical engineering, petrochemicals, electronics, etc.), continuous expansion and maintenance of these dictionaries will be required, which was not addressed in the present study.

4) Lack of certification and compliance with information security requirements.

The prototype has not been certified for compliance with industrial cybersecurity standards (IEC 62443, GOST R 56939, FSTEC requirements for critical information infrastructure protection). Data exchange between ERP and ICP via external processing and web services potentially creates vulnerabilities that require separate analysis and protective measures.

5) Limited support for dynamic and fuzzy asset states.

Although elements of fuzzy logic were used in the prototype to handle description uncertainties, full accounting of dynamic characteristics (changes in asset states over time, degradation, transitions between life cycle stages) is implemented only partially and requires further development.

These limitations do not diminish the significance of the obtained results as a conceptual and prototypical foundation, but highlight the need for subsequent stages: full-scale pilot implementation in real production ecosystems, load and security testing, extension to other ERP platforms, and integration with modern large language models to improve text processing robustness.

Список источников / References

1. Антонов И.В., Воронов М.В. Метод автоматизированного построения онтологии предметной области. Моделирование и анализ данных. 2011. № 1. С. 116–130.
Antonov, I. V., & Voronov, M. V. (2011). Metod avtomatizirovannogo postroeniya ontologii predmetnoi oblasti [Method for automated construction of a domain ontology]. Modelirovanie i analiz dannykh, (1), 116–130. (In Russ.).
2. Дли М.И., Черновалова М.В., Соколов А.М., Моргунова Э.В. Нечеткая динамическая онтологическая модель для поддержки принятия решений по управлению энергоемкими системами на основе прецедентов. Прикладная информатика. 2023. Т. 18. № 5 (107). С. 59–76.
Dli, M.I., Chernovalova, M.V., Sokolov, A. M., & Morgunova, E.V. (2023). Fuzzy dynamic ontological model for decision support of energy-intensive systems management based on precedents. *Journal of Applied Informatics*, 18 (5), 59–76. (In Russ.). <https://doi.org/10.37791/2687-0649-2023-18-5-59-76>
3. Каленов О.Е. Унифицированные цифровые бизнес-модели участников платформ и экосистем. Вестник Российского экономического университета имени Г.В. Плеханова. 2022, (3), 52–59. <https://doi.org/10.21686/2413-2829-2022-3-52-59>



- Kalenov, O.E. (2022). Unified digital business-models of platform and ecosystem participants. *Vestnik of the Plekhanov Russian University of Economics*, (3), 52–59. (In Russ.). <https://doi.org/10.21686/2413-2829-2022-3-52-59>
4. Куравский, Л.С., Одинцов, Д.А., Михайловский, М.А. (2025). Эволюционные алгоритмы подбора запросов и проверки корректности ответов интеллектуальных ассистентов. *Моделирование и анализ данных*, 15(4), 7–26. <https://doi.org/10.17759/mda.2025150401>
Kuravsky, L.S., Odintsov, D.A., & Mikhailovsky, M.A. (2025). Evolutionary algorithms to generate prompts and verify responses of intelligent assistants. *Modelling and Data Analysis*, 15(4), 7–26. (In Russ.). <https://doi.org/10.17759/mda.2025150401>
 5. Поляков Б.Ю. Семантический анализ результатов выполнения тестовых заданий. *Моделирование и анализ данных*. 2025. Т. 15. № 4. С. 156–164.
Polyakov, B.Y. (2025). Semantic analysis of test responses using synthetic data generation. *Modelling and Data Analysis*, 15(4), 156–164. (In Russ.). <https://doi.org/10.17759/mda.2025150410>
 6. Фомин И.Н. Применение онтологических моделей и баз знаний для разработки киберфизических систем в соответствии с референтной моделью RAMI 4.0 *Прикладная информатика*. 2025. Т. 20. № 4 (118). С. 110–131. DOI: 10.37791/2687-0649-2025-20-4-110-131
Fomin, I.N. (2025). Application of ontological models and knowledge bases for the development of cyber-physical systems in accordance with the RAMI 4.0 reference model. *Journal of Applied Informatics*, 20 (4), 110–131. (In Russ.). <https://doi.org/10.37791/2687-0649-2025-20-4-110-131>
 7. Awad, R., Heppner, G., Roennau, A., & Bordignon, M. (2016). ROS engineering workbench based on semantically enriched app models for improved reusability. In *2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)*, 1–9. <https://doi.org/10.1109/ETFA.2016.7733581>
 8. Bader, S.R., Grangel-González, I., Nanjappa, P., Vidal, M.-E., & Maleshkova, M. (2020). A knowledge graph for Industry 4.0. In *The Semantic Web: ESWC 2020 Satellite Events*, 1–16. https://doi.org/10.1007/978-3-030-62327-2_1
 9. Megow, J. (2020). *Reference architecture models for Industry 4.0, smart manufacturing and IoT*. PaiCE.
 10. Müller, J.M. (2019). Antecedents to digital platform usage in Industry 4.0 by established manufacturers. *Sustainability*, 11 (4), Article 1121. <https://doi.org/10.3390/su11041121>
 11. Pauli, T., Fietl, E., & Matzner, M. (2021). Digital industrial platforms. *Business & Information Systems Engineering*, 63, 181–190. <https://doi.org/10.1007/s12599-020-00681-w>
 12. Reimers, N. & Gurevych, I. (2019). *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks*. Proceedings of EMNLP-IJCNLP, 3982–3992. <https://doi.org/10.18653/v1/D19-1410>
 13. Rong, K., Lin, Y., Du, W., & Yang, S. (2023). Business ecosystem-oriented business model in the digital era. *Technology Analysis & Strategic Management*. <https://doi.org/10.1080/09537325.2023.2191743>
 14. Wang, L., Yang, N., Huang, X., Jiao, B., Yang, L., Jiang, D., Majumder, R., & Wei, F. (2022). *Text Embeddings by Weakly-Supervised Contrastive Pre-training (E5)*. <https://doi.org/10.48550/arXiv.2212.03533>
 15. Xiao, S., Liu, Z., Zhang, P., & Muennighoff, N. (2024). *C-Pack: Packed Resources for General Text Embedding (BGE)*. *arXiv:2309.07597v5 [cs.CL]* 24 Sep 2024.



16. Yahya, M., Shah, S. H., Khattak, H. A., Shafiq, M. O., & Khan, M. A. (2021). Semantic web and knowledge graphs for Industry 4.0. *Applied Sciences*, 11(11), Article 5110. <https://doi.org/10.3390/app11115110>

Информация об авторе

Фомин Игорь Николаевич, кандидат технических наук, старший научный сотрудник, Институт проблем точной механики и управления — обособленное структурное подразделение Федерального государственного бюджетного учреждения науки Федерального исследовательского центра «Саратовский научный центр Российской академии наук» (ИПТМУ РАН), доцент кафедры системного анализа и автоматического управления, Федеральное государственное бюджетное образовательное учреждение высшего образования «Саратовский национальный исследовательский государственный университет имени Н.Г. Чернышевского» (ФГБОУ ВО СГУ), Саратов, Российская Федерация, ORCID: <https://orcid.org/0000-0003-3491-6888>, e-mail: ignik16@yandex.ru

Information about the author

Igor N. Fomin, Candidate of Engineering Sciences, Senior Researcher at the Institute of Precision Mechanics and Control Problems, Saratov Scientific Center of the Russian Academy of Sciences, Saratov, Russian Federation; Associate Professor at Yuri Gagarin State Technical University of Saratov, Saratov, Russian Federation, ORCID: <https://orcid.org/0000-0003-3491-6888>, e-mail: ignik16@yandex.ru

Конфликт интересов

Авторы заявляют об отсутствии конфликта интересов.

Conflict of interest

The authors declare no conflict of interest.

Поступила в редакцию 27.03.2026

Поступила после рецензирования 10.08.2026

Принята к публикации 12.08.2026

Опубликована 30.09.2026

Received 2026.03.27

Revised 2026.08.10

Accepted 2026.08.12

Published 2026.09.30