

Труды МАИ. 2025. № 142
Trudy MAI. 2025. No. 142. (In Russ.)

Научная статья

УДК 53.087.92

URL: <https://trudymai.ru/published.php?ID=185107>

EDN: <https://www.elibrary.ru/LZPPJG>

МОДЕЛИРОВАНИЕ АРХИТЕКТУРЫ КОГНИТИВНОГО ИНТЕРНЕТА ВЕЩЕЙ ДЛЯ ИНФОРМАЦИОННО- ИЗМЕРИТЕЛЬНЫХ И АВТОМАТИЗИРОВАННЫХ СИСТЕМ

Андрей Сергеевич Сомов

Сколковский Институт Науки и Технологий,

Москва, Россия

a.somov@skoltech.ru

Аннотация. Предложен подход к моделированию архитектуры когнитивного интернета вещей. В рамках исследования определен набор базовых принципов, моделирование структурных элементов и функциональных возможностей. Это привело к созданию функциональной архитектуры, способной представлять объекты реального мира и их виртуализацию, виртуальные объекты и виртуальные составные объекты. В рамках когнитивного цикла, в котором знания извлекаются из наблюдения за внешней средой, которая постоянно развивается, решения/действия определяют будущее поведение систем в рамках сервисного запроса. Проведена оценка времени отклика предложенного архитектурного решения на сервисные запросы, включающие разное количество виртуальных составных объектов, что позволит применять архитектуру когнитивного интернета вещей для моделирования разнородных систем

мониторинга с необходимым количеством датчиков, и измерительных систем и сервисов.

Ключевые слова: интернет вещей, информационно-измерительные системы, набор датчиков, интеллектуальные системы, мониторинг

Для цитирования: Сомов А.С. Моделирование архитектуры когнитивного интернета вещей для информационно-измерительных и автоматизированных систем

// Труды МАИ. 2025. № 142. URL: <https://trudymai.ru/published.php?ID=185107>

Original article

MODELING THE ARCHITECTURE OF THE COGNITIVE INTERNET OF THINGS FOR INFORMATION AND MEASUREMENT SYSTEMS AND AUTOMATED SYSTEMS

Andrey S. Somov

Skolkovo Institute of Science and Technology,

Moscow, Russia

a.somov@skoltech.ru

Abstract. An approach to modeling the architecture of the cognitive Internet of Things (IoT) is proposed. Within the framework of the study, a set of basic principles, modeling of structural elements and functional capabilities was determined. This has led to the creation of a functional architecture capable of representing real-world objects and their virtualization, virtual objects and virtual composite objects. Within the framework of the cognitive cycle, in which knowledge is extracted from observing the external environment,

which is constantly evolving, decisions/actions determine the future behavior of systems within the framework of a service request. From the point of view of information and measurement systems, this means that any real object can be represented as a virtual object (VO), which can be used or reused in the required context according to the decision of the cognitive IoT architecture. All VOs can be represented as a composite VO (CVO) upon service request. However, before compiling the CVO, the IoT architecture checks the availability of relevant CVO already compiled in the registry to quickly prepare for a service request. Practically, the proposed cognitive IoT architecture helps to exclude a person from the process of service operation, analysis, and maintenance of the system. This is especially important when scaling information and measurement and automated systems. A human is only needed to generate a service request if it cannot be generated automatically. In addition, the proposed architecture allows for a conceptual transition from the level of modeling and networking of physical objects to the level of service provision. Thus, any object and object can be presented as a service and support the "servitization" model. The evaluation of the response time of the proposed architectural solution to service requests involving a different number of virtual composite objects has been carried out, which will allow using the architecture of the cognitive Internet of Things to model heterogeneous monitoring systems with the necessary number of sensors and measuring systems and services.

Keywords: thermal catalytic hydrogen sensor, platinum group catalysts, low-temperature catalytic combustion, reaction self-initiation temperature

For citation: Somov A.S. Modeling the architecture of the cognitive internet of things for information and measurement systems and automated systems. *Trudy MAI*. 2025. No. 142. (In Russ.). URL: <https://trudymai.ru/eng/published.php?ID=185107>

Введение

Интернет вещей, впервые предложенный Кевином Эштоном в качестве названия презентации в 1999 году, представляет собой технологическую революцию, которая вводит нас в новую эпоху повсеместного подключения, вычислений и коммуникации [1]. Развитие Интернета вещей зависит от динамичных технических инноваций в ряде областей, от беспроводных датчиков до нанотехнологий [2]. Чтобы эти новаторские инновации переросли из идей в конкретные продукты или приложения диагностики и контроля, за последнее десятилетие мы стали свидетелями усилий академического сообщества, поставщиков услуг, сетевых операторов, организаций по разработке стандартов и т.д. по всему миру. Например, методы моделирования Интернета вещей применяются в аэрокосмической промышленности [3] и наземных радионавигационных системах [4]. В частности, летательные и космические аппараты являются набором электронных средств и средств связи, которые представляют собой «вещи» в контексте Интернета вещей, а генерируемые данные могут использоваться в комплексе для задач предиктивной аналитики. Технически, большая часть внимания была сосредоточена на таких аспектах, как коммуникация, вычисления, подключение к интернету и т.д., которые действительно являются очень важными темами. Однако, без всесторонних когнитивных способностей Интернет вещей подобен неуклюжему стегозавру: сплошные мускулы, но никаких интеллектуальных способностей. Чтобы реализовать его потенциал для задач диагностики и контроля, а также справиться с растущими вызовами, мы должны учитывать когнитивные способности и наделить эту парадигму интеллектом высокого уровня.

Текущие исследования в области Интернета вещей, в основном, сосредоточены на том, как предоставить возможность обычным объектам или «вещам» самим делать измерения в окружающем мире, а также подключать их к сети для обмена данными [5]. Для научного сообщества становятся очевидными проблемы разнородности датчиков, проблемы нехватки функционала измерительных систем/сервисов и увеличения генерируемых данных [6]. Кроме этого, особенно остро встает проблема питания автономных датчиков - проявляется необходимость сбора альтернативной энергии малых мощностей. Стоит отметить, что по мере интеграции каждого технологического стека в парадигму Интернета вещей находится ряд научных задач, которые, зачастую, необходимо решать в комплексе непосредственно парадигмы Интернета вещей. В предлагаемой диссертации доказано, что одних подключений к сети огромного количества разнородных объектов недостаточно для реализации эффективных приложений. Кроме этого, в том числе в контексте 4-й технологической революции [7], автономные объекты и объекты общие для нескольких приложений должны обладать способностью самостоятельно учиться, мыслить и понимать как физический, так и социальный миры для генерирования автономного управляющего решения. Эта практическая потребность побуждает к моделированию и разработке новой парадигмы – Когнитивного интернета вещей (КИВ), чтобы наделить существующий Интернет вещей «мышлением» для высокоуровневого интеллекта.

В рамках сетевой парадигмы КИВ [8] (физические/виртуальные) вещи, объекты и облачные сервисы взаимосвязаны и ведут себя как агенты [9], с минимальным вмешательством человека. Вещи взаимодействуют друг с другом в соответствии с контекстно-зависимым циклом восприятия-действия, используют методологию

понимания путем построения знаний, извлекая информацию из физической среды и сохраняя усвоенную семантику и/или знания в различных базах данных. Данный подход позволяет адаптироваться к изменениям или неопределенностям с помощью ресурсосберегающих механизмов принятия решений, имея в виду следующее:

1. соединяя физический мир (с объектами, ресурсами и т.д.) и виртуальный мир (с человеческими потребностями, социальным поведением и т.д.), вместе с самими собой, чтобы сформировать интеллектуальную киберфизическую систему;

2. обеспечение разумного распределения и использования ресурсов, автоматической работы вещей и сети [10], а также предоставления интеллектуальных услуг.

Целью работы является моделирование архитектуры Когнитивного Интернета Вещей (КИВ) с определением набора базовых принципов, моделирование структурных элементов и функциональных возможностей.

1. Обзор литературы

Концепция Интернета вещей [11] заключается в том, чтобы позволить вещам подключаться в любое время, в любом месте к чему угодно и к кому угодно, в идеальном случае используя любую сеть и любой сервис. Виртуальные представления объектов реального мира (ОРМ) содержат информацию об их идентификации и описании или ссылаются на нее. ОРМ могут передавать информацию о себе и могут получать доступ к информации, предоставляемой другими участниками, через Интернет. Кроме того, существуют различные технологии, которые рассматриваются при реализации Интернета вещей, такие как RFID, сенсорные сети, семантические технологии, межмашинная связь (M2M).

Концепция Web of Things (WoT) предполагает интеграцию интеллектуальных "вещей", их сервисов и аналогов в сети, что может быть рассмотрено как расширенная концепция Интернета вещей [12]. Таким образом, концепция WoT увеличивает потребность в "интеллекте", который обеспечит возможность подключения, взаимодействия в сетях, функциональной совместимости и актуальности между физическим миром и соответствующими ресурсами виртуального мира.

В отличие от (обычного) Интернета Вещей (ИВ) [8][11], исследования Когнитивного Интернета Вещей (КИВ) очень ограничены. В [13] представлена структура когнитивного управления, позволяющая IoT лучше поддерживать устойчивое развитие умного города, где когнитивное управление в основном относится к автономному выбору наиболее релевантных объектов для данного приложения. В [14] КИВ рассматривается как современный интернет вещей, интегрированный с когнитивными и кооперативными механизмами для повышения производительности и достижения интеллекта, где когнитивный процесс состоит из трехслойного когнитивного кольца. Также приведены исследования, направленные на эффективное принятие решений в рамках Интернета вещей [15].

Первые попытки виртуализации объектов реального мира (ОРМ) или "вещей" были связаны с использованием радиочастотных идентификаторов (RFID) [16], которые могли захватывать только "сырые" данные. Следующие шаги были сделаны с обработкой собранных данных [17]. В настоящее время виртуальные объекты (ВО), то есть виртуальные представления объектов/вещей, как правило, становятся "умнее", обогащая свои модели функциями когнитивного управления и пользовательской информацией [18]. Несмотря на то, что было проведено много исследований с точки

зрения виртуализации в области Интернета вещей, требуется гораздо больше исследований, чтобы понять, как реальные объекты и соответствующие им ВО могут быть подключены и корректно взаимодействовать, или как эти концепции могут быть использованы и реализованы.

2. Основные принципы высокоуровневой и функциональной архитектуры когнитивного интернета вещей

Архитектура Когнитивного Интернета Вещей (КИВ) определяет набор базовых принципов, структурных элементов, функциональных возможностей и инструкций для построения информационно-измерительных и автоматизированных систем. Определение этих архитектурных принципов было выполнено несколькими способами: (а) были рассмотрены требования типичных приложений в рамках указанных систем с целью определения ключевых принципов, которые привели к формулированию ряда технических требований; (б) рассмотрены технические требования, формирующиеся по архитектурному принципу ``сверху вниз"; и (в) были приняты во внимание требования, вытекающие из подхода ``снизу вверх" (т.е. путем определения и спецификации набора вариантов приложений) [19]. Это привело к созданию функциональной архитектуры, способной представлять Объекты Реального Мира (ОРМ) и их виртуализацию в рамках КИВ. Центральное место в разработке определения функциональной архитектуры КИВ занимают несколько основных принципов:

- Идентификация и роли заинтересованных сторон. Базовая архитектура выделяет основные приложения функциональной архитектуры и соответствующие бизнес роли. Они используются для того, чтобы

полностью описать функционирование системы КИВ с точки зрения функциональных и эксплуатационных потребностей, а также бизнеса.

- Виртуализация и компоновка объектов. Объекты Реального Мира (ОРМ) могут быть представлены в КИВ как Виртуальные Объекты (ВО). Они позволяют представить ОРМ в функционально обогащающей среде. Простые ВО могут быть агрегированы и объединены для создания новых Виртуальных Составных Объектов (ВСО), которые расширяют и обобщают функциональные возможности ОРМ, часто в контексте выполнения запроса пользователя и предоставления сервиса.
- Сегментация и агрегирование функций. Объекты структурированы по набору уровней, каждый уровень объединяет функциональные возможности, предлагаемые определенным типом сущностей КИВ: уровень ВО собирает все функциональные возможности, предоставляемые виртуальными объектами, а также сущностями и функциональными возможностями системы КИВ, необходимыми для поддержки и использования ВОв. На вершине уровня ВО КИВ описывает функциональные возможности, связанные с СО, и то, как они могут быть связаны друг с другом и с объектами других уровней (а именно ВОв и функционал предоставления сервисов). На самом верхнем уровне базовая архитектура учитывает все функции, обеспечивающие предоставление сервисов, и их связь с приложениями и ВСО. На каждом уровне архитектура КИВ предусматривает увеличение числа функциональных возможностей и

систематизированных сущностей, используемых для поддержки архитектуры.

- Функциональный и систематический взгляд на объекты. Объекты и составные блоки архитектуры КИВ разработаны таким образом, чтобы предлагать конкретные функции, которые будут использоваться для поддержки сервисов. Любой отдельный объект будет предоставлять интерфейсы и функции, которые поддерживают интеграцию объекта в систему (систематическое представление). Каждый базовый компонент будет поддерживать эти два аспекта, функциональный и систематический, что позволит поддерживать сервисы конечных пользователей и должным образом управлять объектом в контекстных рамках.
- Когнитивность. Познание является еще одним краеугольным камнем архитектуры КИВ. В рамках КИВ оно структурировано как когнитивный цикл, в котором знания извлекаются из наблюдения за внешней средой (т.е. событиями/данными реального мира), которая постоянно развивается, и решения/действия определяют ее будущее поведение на основе трех критериев, а именно: (а) созданные знания, (б) другие цели, а также (в) требования, направленные на оптимизацию производительности.

Стоит отметить, что важным элементом архитектуры КИВ является безопасность. Данный элемент не является вкладом данного исследования.

Другой важной целью архитектуры КИВ является предоставление приложениям доступа к хорошо структурированному и богатому набору сервисных функций. Чтобы удовлетворить это требование, архитектура КИВ использует

концепцию, называемую принципом сервитизации, который необходим для создания связи между (физическим) продуктом и набором услуг и расширенных функциональных возможностей, которые расширяют, дополняют и повышают ценность самого продукта. То есть архитектура структурирована таким образом, чтобы на раннем этапе идентифицировать набор универсальных сервисов, вытекающих из функций, предлагаемых ВО и ВСО или сущностями архитектуры. Другим способом рассмотрения этого принципа является возможность создания набора сервисов на основе ВСОв или даже ВОв.

На рисунке 1 изображена блок-схема архитектуры КИВ. Взаимодействие с архитектурой КИВ инициируется посредством сервисного запроса, сгенерированного с целью активации потока данных от объектов ИВ и их непрерывной обработки для поддержки конечного пользователя / ИКТ-приложения с набором процессов, отслеживающих контекст и выдающих предупреждения при выполнении определенных условий. Такие процессы, предоставленные сервисными шаблонами, организуются и привязываются к соответствующим ВО'ам с использованием функциональности КИВ.

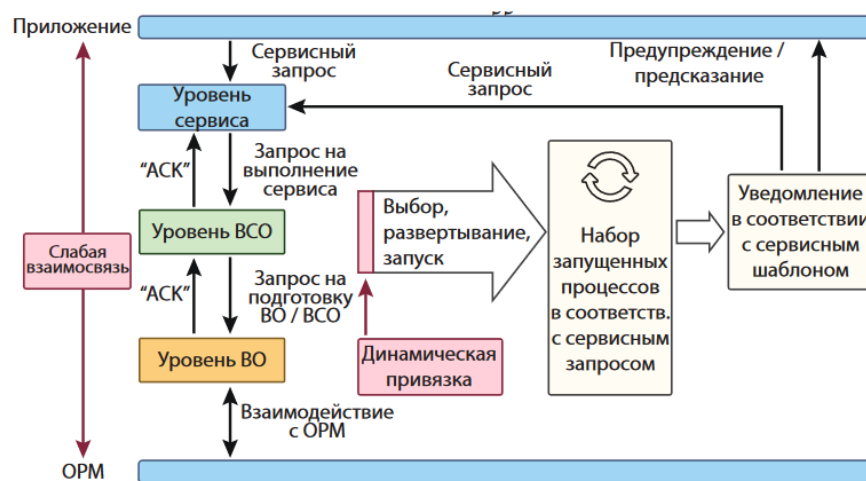


Рисунок 1. Блок-схема архитектуры КИВ.

На блок-схеме показано взаимодействие между уровнями, направленными на активацию набора процессов, поддерживаемых ИВ. Слабая связь между потребностями приложения и объектами ИВ, которые используются для их удовлетворения, реализуется с помощью динамической привязки, способной выбирать наиболее релевантные из них по мере изменения ситуации/контекста, окружающего приложение/пользователя, отправляющего запрос на сервис. На рисунке 1 также показаны взаимодействия между уровнями архитектуры КИВ, после запроса на сервис, который после выбора шаблона сервиса генерирует запрос на выполнение сервиса, используемый для генерации соответствующих ВСО. В свою очередь ВСО зависят от функций базовых ВОв. Данные взаимодействия приводят к набору запущенных процессов, которые, будут генерировать уведомления, доставляемые в приложение, на протяжении всего срока их службы.

На основе блок-схемы архитектуры КИВ разработана функциональная архитектура КИВ, показанная на рисунке 2 включает в себя в основном функциональные блоки, которые обозначают функциональные возможности. Эти возможности, необходимые для правильной работы архитектуры КИВ. Базовые интерфейсы необходимы для связи нескольких участников с архитектурой КИВ и обмена информацией между функциональными блоками.

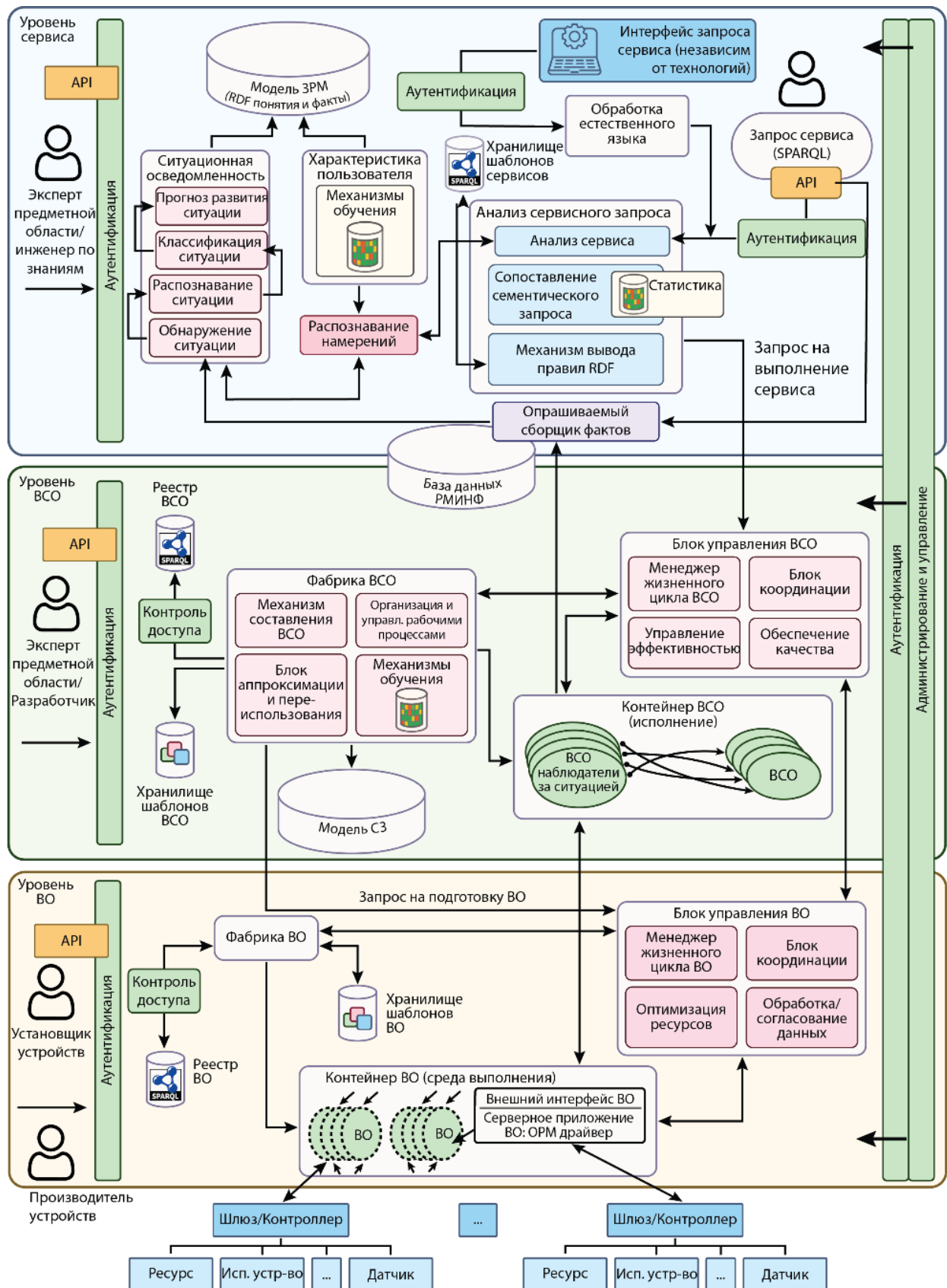


Рисунок 2. Функциональная архитектура КИБ.

Работа архитектуры КИВ начинается с подачи запроса сервиса на *уровне сервиса*. Запрос сервиса подается от технического или нетехнического пользователя архитектуры КИВ. В случае, если лицо, запрашивающее услугу, является нетехническим пользователем, используется пользовательский интерфейс для помощи взаимодействия с архитектурой КИВ. Однако пользовательский интерфейс остается за пределами спецификации архитектуры КИВ. В случае, если лицо, запрашивающее услугу, является техническим пользователем, архитектура КИВ предоставляет спецификацию API. Обработка естественного языка: этот функциональный блок применим только в том случае, если пользователь архитектуры КИВ является нетехническим пользователем, который отправляет запрос на естественном языке, набранный в пользовательском интерфейсе, либо потенциально с помощью программного обеспечения для распознавания речи. Функциональный блок преобразует эти нетехнические пользовательские запросы и инструкции на человеческом языке в формальный запрос SPARQL. Анализ сервисных запросов: этот функциональный блок состоит из трех функциональных подблоков. Анализ сервиса является отправной точкой блока. Он получает запрос SPARQL и запрашивает статус текущей ситуации. Затем он пересылает параметры запроса и ситуации (параметры ситуации) в блок сопоставления семантических запросов. Блок сопоставления семантических запросов включает в себя улучшения семантического выравнивания/обучения в качестве потенциальной предварительной обработки для стандартного SPARQL-сопоставления запроса с сервисом. Данный блок необходим, поскольку на практике редко существует идеальное соответствие запроса набору шаблонов из-за неполноты модели предметной области, охватывающей все

семантические концепции, используемые в запросах на сервис. Результатом работы механизма вывода правил RDF является логическая совокупность критериев выбора BCO, называемая запросом на выполнение сервиса, который должен быть передан на уровень BCO для выполнения сервиса. Запрос на выполнение сервиса является конечным результатом анализа запроса на сервис и включает в себя, помимо критериев отбора BCO, критерии соглашения об уровне сервиса, которые выражают требования к качеству и критерии стоимости. Хранилище шаблонов сервисов: оно содержит семантически запрашиваемую коллекцию шаблонов служб, предоставленную в хранилище экспертом по предметной области/инженером по знаниям. Репозиторий шаблонов служб содержит множество неявных знаний реального мира, и поэтому выражение их в виде правил RDF можно рассматривать как виртуальное расширение модели знаний реального мира. Распознавание намерений: этот блок включает в себя когнитивную функциональность, которая используется для определения намерений пользователя (например, что интересует пользователя?). Это помогает определить “Цели мониторинга” (специфичные для конкретного приложения), необходимые для подблока “Обнаружение ситуации” блока “Информирование о ситуации”. Характеристика пользователя: включает в себя определение ряда фактов, касающихся пользователя-человека, включая пользовательский контекст, профиль. Ситуационная осведомленность: этот блок отвечает за создание знаний реального мира, которые затем сохраняются в модели знаний реального мира. Процесс понимания ситуации генерируется логической последовательностью и состоит из следующих шагов/подблоков: (i) обнаружение ситуации, (ii) распознавание ситуации, (iii) классификация ситуации и (iv)

прогнозирование развития ситуации. Познание добавляет элемент интеллектуальности, который помогает распознавать ситуацию и тем самым осуществлять действие. Каждый из вышеперечисленных подблоков соответствует циклу познания с точки зрения восприятия (ввода) - понимания (обработки) – адаптации (реагирования), настроенной на конкретные цели. Модель Знаний Реального Мира (ЗРМ): для представления внутренних данных знаний реального мира мы предполагаем, что “знания” могут быть записаны в графиках RDF, и поэтому хранилище моделей ЗРМ можно использовать для запоминания отражения правил поведения реального мира в архитектуре КИВ.

Уровень ВСО получает от уровня сервиса запрос на выполнение сервиса. Входной точкой на уровне ВСО является Блок управления ВСО. Блок управления ВСО: Входной точкой в блоке управления ВСО после запроса на выполнение сервиса является Менеджер жизненного цикла ВСО, который рассматривается как интеллектуальный блок мониторинга, отслеживающий изменяющиеся состояния набора запущенных ВСО. Функциональный блок координации предусмотрен в качестве средства разрешения конфликтов. Например, когда ВСО используются в контексте запросов на выполнение нескольких сервисов, потенциально разные цели сервиса могут влиять на одни и те же ресурсы конфигурации архитектуры КИВ. Управление эффективностью намерено гарантировать надлежащую работу не только уровня ВСО (функциональные блоки и ВСО), но и уровня ВО (функциональные блоки и ВО) с точки зрения соответствия определенным пороговым значениям ключевых показателей эффективности. Фабрика ВСО: в то время как основной задачей блока управления ВСО является управление ресурсами, связанными с

выполнением / средой выполнения, фабрика ВСО занимается созданием новых экземпляров ВСО. Модель Системных Знаний (СЗ): СЗ, которые создаются с помощью механизмов обучения, хранятся в модели СЗ. Хранилище шаблонов ВСО: этот блок содержит (семантически и) функционально запрашиваемый набор шаблонов ВСО. Шаблоны ВСО хранятся экспертом/разработчиком анализа данных. Реестр ВСО: содержит метаданные для каждого развернутого экземпляра ВСО, которые сохраняются в течение определенного периода времени. Контейнер ВСО: обозначает фактическую среду выполнения экземпляров ВСО. Экземпляры ВСО и контейнер ВСО отслеживаются, контролируются и управляются блоком управления ВСО. Наблюдатели за ситуацией: это специальные ВСО, которые занимаются конкретной задачей наблюдения за событиями посредством обработки ВСО, как особенно значимыми и относящимися к “ситуации”, в которой находится конкретный человек или сервис. Цель заключается в том, чтобы в конечном итоге подготовленные наблюдатели были применимы для многих приложений и ситуационных аспектов. Это имеет большое значение, поскольку фокусирует знания реального мира более на те наблюдения, которые наиболее актуальны для пользователя и сервисов, которые они определяют или запускают. Наблюдатели обеспечивают ввод данных во время выполнения для информирования о ситуации через запрашиваемый быстрый сборщик. Стоит отметить, что ВСО может включать только один ВО, но, тем не менее, эта сущность считается ВСО, если она включает в себя несколько когнитивных функций, которые недоступны в отдельном ВО.

Виртуализация используется на всех уровнях архитектуры КИВ. Однако *уровень ВО* является основополагающим уровнем для виртуализации в том смысле,

что уровень ВО отвечает за виртуализацию данных датчиков (и исполнительных устройств) для любых сервисных нужд. Помимо возможностей виртуализации, уровень ВО включает в себя дополнительные возможности управления для контроля и распоряжения объектами виртуального и реального мира. Уровень ВО получает от уровня ВСО запрос на подготовку ВО. Этот запрос отправляется с фабрики ВСО в блок управления ВО. Блок управления ВО: Подобно блоку управления ВСО, блок управления ВО также занимается управлением экземплярами ВО во время их работы. Входной точкой в блоке управления ВО после запроса на выполнение ВО является менеджер жизненного цикла ВО. Менеджер жизненного цикла ВО можно рассматривать как интеллектуальную функцию мониторинга, отслеживающую информацию о состоянии жизненного цикла отдельных ВОв в данный момент. Функциональный блок координации необходим для обеспечения нормальной работы во время выполнения посредством разрешения конфликтов и нестабильной работы, возникающих из-за различных одновременно вызываемых ВОв, которые хотят управлять одними и теми же аспектами конфигурации архитектуры КИВ и/или конфигурацией датчика/исполнительного устройства/ресурса. Оптимизация ресурсов реализуется с помощью когнитивных функций работы базовых датчиков, исполнительных механизмов и ресурсов, например, за счет снижения энергопотребления. Обработка/ согласование данных обеспечивает управление данными и обеспечивает качество данных, например, путем интерполяции недостающих данных с использованием методов машинного обучения. Блок управления ВО контролируется блоком управления ВСО. Фабрика ВО: в то время как блок управления ВО связан с фазой выполнения, фабрика ВО рассматривает фазу

создания / начальной загрузки. Хранилище шаблонов ВО: этот блок содержит семантически запрашиваемый набор шаблонов ВО. Шаблоны ВО хранятся производителем устройства, который представляет промышленную организацию, создающую устройства (датчики, исполнительные устройства, сервисы) и описывающую их функциональные возможности в таких шаблонах. Реестр ВО: реестр ВО содержит метаданные для каждого установленного ВО, которые сохраняются в течение определенного периода времени. Реестры ВО хранят семантически обогащенные данные, которые используются для описания ВО, чтобы быть доступными в любое время и из любого места. Контейнер ВО: Контейнер ВО обозначает среду выполнения экземпляров ВО. Экземпляры ВО и контейнер ВО отслеживаются, контролируются и управляются блоком управления ВСО через блок управления ВО. Данные передаются из контейнера ВО в контейнер ВСО.

3. Результаты и обсуждения

В рамках анализа функционала когнитивной архитектуры и времени динамического создания ВСО'в по сервисному запросу была полностью реализована функциональная архитектура КИВ на примере моделирования контроля температуры в здании.

Уровень сервиса: в текущей реализации в качестве входящего запроса используется запрос на управление приборами с целью поддержания комфортной температуры в здании. Этот запрос генерирует требование ВСО, состоящее из датчиков температуры, влажности и освещенности, расположенных в определенной пространственной области. Сервисный запрос осуществляется пользователем с помощью графического интерфейса пользователя. Затем запрос передается в блок

анализа сервисного запроса, который сопоставляет полученный запрос с заранее определенными шаблонами сервисов. Далее сформированные требования к ВСО передаются на уровень ВСО архитектуры.

Уровень ВСО: при получении запроса на сервис блок управления ВСО выполняет поиск в реестре ВСО существующего ВСО, который может предоставить запрошенный сервис. Если такой ВСО недоступен, то он выполняет поиск в блоке управления ВО, чтобы найти соответствующие ВО, способные обеспечить требуемые функциональные возможности в дополнение к удовлетворению ограничений, сгенерированных запросом. ВО, удовлетворяющие всем требованиям, объединяются для создания ВСО, который затем может самостоятельно предоставлять требуемый сервис. Например, запрос на регулирование температуры включает поиск ВСО с указанием температуры в здании. Если ВСО, который может предоставить значения температуры, недоступен, блок управления выполняет поиск в реестрах ВО. Затем он формирует ВСО на основе данных ВО'в, полученных от блока управления ВО датчиков температуры, а также вентилятора, который используется для регулирования температуры. ВСО содержит метаданные ВО, а также сведения для подключения к ВО и ОРМ по очереди.

Уровень ВО: в этой работе мы представляем наш ОРМ, как ВО в платформе Xively [22]. При регистрации ОРМ проводится их дополнительное описание контекстной информацией об объекте, такой как местоположение, функциональность, единицы измерения, статус и сведения о владельце. Все ВО хранятся в Xively в машиночитаемых форматах данных XML, и, по сути, представляют собой веб-ресурсы, доступ к которым можно получить с помощью команд RESTful. Виртуальное

представление реальных объектов обеспечивает совместимость между различными технологиями. Xively позволяет подключаться к ВО в любое время из любого места, и, кроме того, при использовании веб-сервисов REST (англ. REpresentational State Transfer) и MQTT (англ. message queuing telemetry transport) сложность реализации канала связи становится низкой. В данной реализации используются скрипты на языке Python для отправки данных, собранных с датчиков, в Xively канал, который служит источником информации для реестра ВО. Исполнительные устройства прослушивают Xively канал с другим номером, который предоставляет управляющую информацию для приведения в действие устройств.

Модель знаний о реальном мире: модель знаний о реальном мире должна быть способна фиксировать и сохранять знания о событиях и решениях, которые являются прямыми последствиями постоянно меняющихся ситуаций в окружающей среде. Это позволяет системе принимать обоснованные решения и определяет будущее поведение системы. Данные, которые поступают от ОРМ в ВСО, передаются в модель знаний реального мира. Когнитивные способности в фабрике ВСО разделены на «рассуждение» и «обучение». «Рассуждение» содержит все ограничения, которые должны быть выполнены, чтобы получить требуемый результат. Например, рассмотрим температурный ВСО, который постоянно предоставляет информацию о значениях температуры в доме. Эта информация передается в систему управления домом (не показана на рисунках), и когда значения превышают установленные пороговые значения, это вызывает повышение температуры, что приводит к включению вентилятора.

Способность «обучения» предоставляет возможность отслеживать модели предпочтений с течением времени, которые используются для составления прогнозов, и с определенной степенью достоверности может автоматизировать процесс регулирования температуры. Преимущество заключается в возможности использовать сильные стороны обоих видов интеллекта. Функционал «рассуждений» реализован как движок, основанный на правилах, Drools. Это платформа интеграции бизнес-логики с открытым исходным кодом для правил, рабочих процессов и обработки событий [23]. В текущей реализации библиотека Drools используется в качестве базы правил для активации различных исполнительных механизмов, когда датчики удовлетворяют правилам. Функционал «обучения» реализован с помощью алгоритмов машинного обучения. При реализации был использован WEKA toolkit [24], библиотека алгоритмов машинного обучения с открытым исходным кодом. Чтобы продемонстрировать функциональность обучения, температурные предпочтения пользователя были записаны с течением времени через пользовательский интерфейс. На основе этой информации и с помощью алгоритма многослойного перцептрона изучаются предпочтения пользователя и используются для прогнозирования предпочтительной температуры в зависимости от времени суток. Таким образом, эта информация позволяет автоматизировать управление приводами, настроенными в соответствии с предпочтениями пользователя и без его вмешательства. В работе достигнута точность 95% при неправильной установке температуры в зависимости от времени суток.

Помимо этого, было проведено моделирование, варьирующее количество ВО в реестре ВО, количество ВО в составе ВСО, а также время, затрачиваемое на поиск по

существующему ВСО и составление нового ВСО. Замечено, что на основе текущей реализации поиск из реестра ВСО занимает меньше времени, чем для создания нового ВСО. Время, затрачиваемое на составление ВСО для каждого сервисного запроса представлено на рисунке 3. При поступлении первого сервисного запроса составляется новый ВСО, а для последующих запросов из реестра подбирается ранее составленный ВСО. Для формирования первого ВСО требуется не менее 300 мс. Четыре кривые отображают тесты с различными параметрами, с точки зрения количества ВО в реестре и количества ВО в ВСО. Таким образом, показано преимущество повторного использования составленных ВСО перед составлением нового ВСО для каждого полученного запроса и интеллектуальный подбор уже имеющихся ВСО для сервисного запроса.

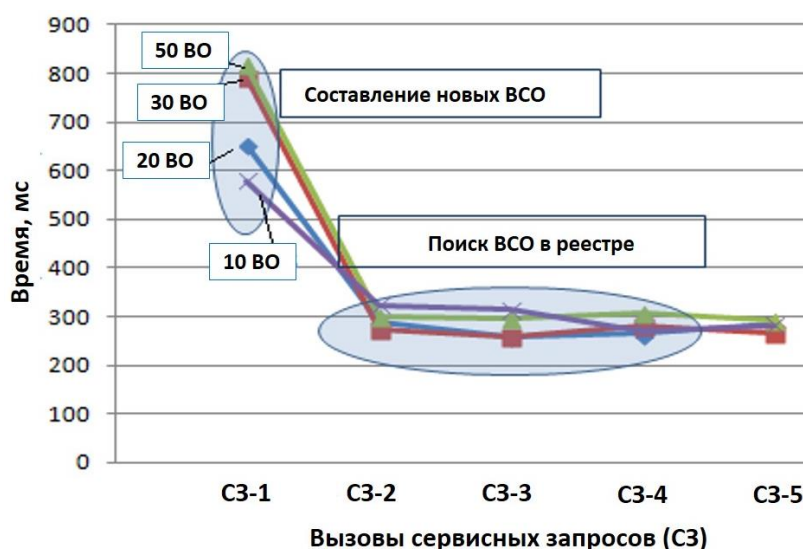


Рисунок 3. Анализ времени сервисного запроса.

Выводы

В статье предложена, теоретически обоснована и экспериментально верифицирована архитектура КИВ. Впервые предложена виртуализация и компоновка объектов реального мира, когнитивные функции, сегментация и

агрегирование функций, функциональный и систематический взгляд на объекты, ситуационная осведомленность и когнитивность.

С точки зрения информационно-измерительных систем это означает, что любой ОРМ может быть представлен в виде ВО, который может быть по решению КИВ использован или переиспользован в требуемом контексте, в том числе это касается таких сложных систем, как летательные и космические аппараты. Все ВО могут быть представлены в качестве ВСО по сервисному запросу. Однако перед составлением ВСО архитектура КИВ проверяет наличие уже составленных релевантных ВСО в реестре для быстрой подготовки к сервисному запросу.

Практически предложенная архитектура КИВ способствует исключению человека из процесса работы сервиса, анализа, и поддержания работы системы. Это особенно важно при масштабировании информационно-измерительных и автоматизированных систем. Человек необходим только для генерации сервисного запроса, если он не может быть сгенерирован автоматическим образом. Помимо этого, предложенная архитектура позволяет концептуально перейти с уровня моделирования и соединения в сеть физических предметов на уровень предоставления сервисов. Таким образом, любой предмет и объект может быть представлен в качестве сервиса и поддерживать модель «сервитизации».

Предложенная архитектура КИВ может быть использована в разнообразных приложениях, включая умный дом/город/транспорт, мониторинг окружающей среды, промышленный мониторинг, в частности мониторинг утечек газов и оценка взрывоопасности среды, контроль объектов энергетического комплекса [20][21],

моделирование в аэрокосмической промышленности [3] и наземных радионавигационных системах [4].

Список источников

1. Касатиков Н.Н., Брехов О.М., Николаева Е.О. Интеграция технологий искусственного интеллекта и интернета вещей для расширенного мониторинга и оптимизации энергетических объектов в умных городах // Труды МАИ. 2023. № 131. URL: <https://trudymai.ru/published.php?ID=175929>. DOI: [10.34759/trd-2023-131-23](https://doi.org/10.34759/trd-2023-131-23)
2. Qazi Mamoon Ashraf, Mohammad Tahir, Mohamed Hadi Habaebi, Jouni Isoaho. Toward Autonomic Internet of Things: Recent Advances, Evaluation Criteria, and Future Research Directions // IEEE Internet of Things Journal. 2023. V. 10, No. 16. P. 14725–14748. URL: <https://doi.org/10.1109/JIOT.2023.3285359>
3. Кузнецова С.В., Семенов А.С. Цифровые двойники в аэрокосмической промышленности: объектно-ориентированный подход // Труды МАИ. 2023. № 131. URL: <https://trudymai.ru/published.php?ID=175930>. DOI: [10.34759/trd-2023-131-24](https://doi.org/10.34759/trd-2023-131-24)
4. Цымбал М.Р., Семичастнов А.Е., Балакин Д.А., Удалов Н.Н. Разработка цифрового двойника наземной радионавигационной системы по принципам модельно-ориентированного проектирования с помощью математической среды моделирования Engee // Труды МАИ. 2024. № 136. URL: <https://trudymai.ru/published.php?ID=180679>
5. Кочкаров Р.А., Балдычев М.Т., Казанцев А.М., Прокопчина С.В., Тимошенко А.В. Алгоритм оценки структурно-функциональной устойчивости и целостности гетерогенной сети передачи данных пространственно-распределенной системы

мониторинга // Труды МАИ. 2024. № 137. URL:
<https://trudymai.ru/published.php?ID=181887>

6. P. Vlaceas et al. Enabling smart cities through a cognitive management framework for the internet of things // IEEE Communications Magazine. 2013. V. 51, No. 6. P. 102-111. URL: <https://doi.org/10.1109/MCOM.2013.6525602>

7. Лясковская Е.А. Индустрия 4.0 и устойчивое развитие: от устойчивых бизнес-моделей к цифровой устойчивости // Вестник Южно-Уральского государственного университета. Серия: Экономика и менеджмент. 2021. № 4. С. 74-83. DOI: [10.14529/em210408](https://doi.org/10.14529/em210408)

8. Суриков К.А. Архитектура системы интернета вещей // Международный журнал гуманитарных и естественных наук. 2022. № 3-2. С. 152-155. DOI: [10.24412/2500-1000-2022-3-2-152-155](https://doi.org/10.24412/2500-1000-2022-3-2-152-155)

9. Касатиков Н.Н., Фадеева А.Д., Брехов О.М. Мультиагентная система для контроля объектов энергетического комплекса // Труды МАИ. 2023. № 130. URL: <https://trudymai.ru/published.php?ID=174622>. DOI: [34759/trd-2023-130-23](https://doi.org/10.24412/2500-1000-2023-3-2-152-155)

10. Дорожко И.В., Мусиенко А.С. Модель мониторинга технического состояния сложных устройств с применением искусственного интеллекта // Труды МАИ. 2024. № 137. URL: <https://trudymai.ru/published.php?ID=181885>

11. Daniele Miorandi, Sabrina Sicari, Francesco De Pellegrini, Imrich Chlamtac. Internet of things: Vision, applications and research challenges // Ad Hoc Networks. 2012. V. 10, No. 7. P. 1497–1516. URL: <https://doi.org/10.1016/j.procs.2018.05.042>

12. Z. Benomar, F. Longo, G. Merlino, A. Puliafito. A Cloud-Based and Dynamic DNS Approach to Enable the Web of Things // IEEE Transactions on Network Science and

Engineering. 2022. V. 9, No. 6. P. 3968-3978. URL:
<https://doi.org/10.1109/TNSE.2021.3110003>

13. J.E. Siegel, S. Kumar, S.E. Sarma. The Future Internet of Things: Secure, Efficient, and Model-Based // IEEE Internet of Things Journal. 2018. V. 5, No. 4. P. 2386-2398. URL:
<https://doi.org/10.1109/JIOT.2017.2755620>

14. Q. Wu et al. Cognitive Internet of Things: A New Paradigm Beyond Connection // IEEE Internet of Things Journal. 2014. V. 1, No. 2. P. 129-143. URL:
<https://doi.org/10.1109/JIOT.2014.2311513>

15. Трембач В.М. Модульная архитектура интеллектуальной системы для решения задач интернета вещей // Открытое образование. 2019. № 4. С. 32-43. DOI:
[10.21686/1818-4243-2019-4-32-43](https://doi.org/10.21686/1818-4243-2019-4-32-43)

16. Weinstein R. RFID: a technical overview and its application to the enterprise // IT Professional. 2005. V. 7, No. 3. P. 27–33. URL: <https://doi.org/10.1109/MITP.2005.69>

17. Aberer Karl, Hauswirth Manfred, Salehi Ali. Infrastructure for Data Processing in Large-Scale Interconnected Sensor Networks // International Conference on Mobile Data Management. 2007. P. 198–205. URL: <https://doi.org/10.1109/MDM.2007.36>

18. M. Abbasi, A. Shahraki, J. Prieto, A.G. Arrieta, J.M. Corchado. Unleashing the Potential of Knowledge Distillation for IoT Traffic Classification // IEEE Transactions on Machine Learning in Communications and Networking. 2024. V. 2, P. 221-239. URL:
<https://doi.org/10.1109/TMLCN.2024.3360915>

19. Логиновский О.В., Нестеров М.И., Шестаков А.Л. Применение методов архитектурного подхода в развитии информационной системы крупного вуза //

Вестник Южно-Уральского государственного университета. Серия: Компьютерные технологии, управление, радиоэлектроника. 2013. Т. 13, № 4. С. 123-128.

20. Andrey Somov, Alexander Baranov, Denis Spirjakin. A wireless sensor–actuator system for hazardous gases detection and control // *Sensors and Actuators A: Physical*. 2014. V. 210, P. 157-164. URL: <https://doi.org/10.1016/j.sna.2014.02.025>

21. Филонов О.М., Бестугин А.Р., Киршина И.А., Овчинникова Н.А., Окин П.А. Акустолазерный модуль для обнаружения малогабаритных беспилотных летательных аппаратов в урбанизированной среде // *Датчики и системы*. 2023. № 1 (266). С. 40-45. URL: <https://doi.org/10.25728/datsys.2023.1.6>

22. Xively, Platform for Internet of Thingslkl. URL: <https://xively.com/>

23. Drools, business logic integration platform. URL: <http://www.drools.org/>

24. P. Kotak, H. Modi. Enhancing the Data Mining Tool WEKA // 2020 5th International Conference on Computing, Communication and Security (ICCCS), Patna, India, 2020. P. 1-6.

References

1. Kasatikov N.N., Brekhov O.M., Nikolaeva E.O. Integration of artificial intelligence and the Internet of things for advanced monitoring and optimization of energy facilities in smart cities. *Trudy MAI*. 2023. No. 131. (In Russ.). URL: <https://trudymai.ru/eng/published.php?ID=175929>. DOI: [10.34759/trd-2023-131-23](https://doi.org/10.34759/trd-2023-131-23)
2. Qazi Mamoon Ashraf, Mohammad Tahir, Mohamed Hadi Habaebi, Jouni Isoaho. Toward Autonomic Internet of Things: Recent Advances, Evaluation Criteria, and Future

Research Directions. *IEEE Internet of Things Journal*. 2023. V. 10, No. 16. P. 14725–14748. URL: <https://doi.org/10.1109/JIOT.2023.3285359>

3. Kuznetsova S.V., Semenov A.S. Digital twins in the aerospace industry: an object-oriented approach. *Trudy MAI*. 2023. No. 131. (In Russ.). URL: <https://trudymai.ru/eng/published.php?ID=175930>. DOI: [10.34759/trd-2023-131-24](https://doi.org/10.34759/trd-2023-131-24)

4. Tsymbal M.R., Semichastnov A.E., Balakin D.A., Udalov N.N. Development of a digital twin of a ground-based radio navigation system based on the principles of model-oriented design using the mathematical modeling environment Engee. *Trudy MAI*. 2024. No. 136. (In Russ.). URL: <https://trudymai.ru/eng/published.php?ID=180679>

5. Kochkarov R.A., Baldychev M.T., Kazantsev A.M., Prokopchina S.V., Timoshenko A.V. Algorithm for assessment of structural and functional stability and integrity of heterogeneous data transmission network of spatially distributed monitoring system. *Trudy MAI*. 2024. No. 137. (In Russ.). URL: <https://trudymai.ru/eng/published.php?ID=181887>

6. P. Vlachas et al. Enabling smart cities through a cognitive management framework for the internet of things. *IEEE Communications Magazine*. 2013. V. 51, No. 6. P. 102-111. URL: <https://doi.org/10.1109/MCOM.2013.6525602>

7. Lyaskovskaya E.A. Industry 4.0 and sustainable development: from sustainable business models to digital sustainability. *Vestnik Yuzhno-Ural'skogo gosudarstvennogo universiteta. Seriya: Ekonomika i menedzhment*. 2021. No. 4. P. 74-83. (In Russ.). DOI: [10.14529/em210408](https://doi.org/10.14529/em210408)

8. Surikov K.A. Architecture of the internet of things system. *Mezhdunarodnyi zhurnal gumanitarnykh i estestvennykh nauk*. 2022. No. 3-2. P. 152-155. (In Russ.). DOI: [10.24412/2500-1000-2022-3-2-152-155](https://doi.org/10.24412/2500-1000-2022-3-2-152-155)

9. Kasatikov N.N., Fadeeva A.D., Brekhov O.M. Multi-agent system for monitoring objects of the energy complex. *Trudy MAI*. 2023. No. 130. (In Russ.). URL: <https://trudymai.ru/eng/published.php?ID=174622>. DOI: [34759/trd-2023-130-23](https://doi.org/10.21686/1818-4243-2023-130-23)
10. Dorozhko I.V., Musienko A.S. A model for monitoring the technical condition of complex using artificial intelligence. *Trudy MAI*. 2024. No. 137. (In Russ.). URL: <https://trudymai.ru/eng/published.php?ID=181885>
11. Daniele Miorandi, Sabrina Sicari, Francesco De Pellegrini, Imrich Chlamtac. Internet of things: Vision, applications and research challenges. *Ad Hoc Networks*. 2012. V. 10, No. 7. P. 1497–1516. URL: <https://doi.org/10.1016/j.procs.2018.05.042>
12. Z. Benomar, F. Longo, G. Merlino, A. Puliafito. A Cloud-Based and Dynamic DNS Approach to Enable the Web of Things. *IEEE Transactions on Network Science and Engineering*. 2022. V. 9, No. 6. P. 3968-3978. URL: <https://doi.org/10.1109/TNSE.2021.3110003>
13. J.E. Siegel, S. Kumar, S.E. Sarma. The Future Internet of Things: Secure, Efficient, and Model-Based. *IEEE Internet of Things Journal*. 2018. V. 5, No. 4. P. 2386-2398. URL: <https://doi.org/10.1109/JIOT.2017.2755620>
14. Q. Wu et al. Cognitive Internet of Things: A New Paradigm Beyond Connection. *IEEE Internet of Things Journal*. 2014. V. 1, No. 2. P. 129-143. URL: <https://doi.org/10.1109/JIOT.2014.2311513>
15. Trembach V.M. Modular architecture of an intelligent system for solving the problems of the Internet of Things. *Otkrytoe obrazovanie*. 2019. No. 4. P. 32-43. (In Russ.). DOI: [10.21686/1818-4243-2019-4-32-43](https://doi.org/10.21686/1818-4243-2019-4-32-43)

16. Weinstein R. RFID: a technical overview and its application to the enterprise. *IT Professional*. 2005. V. 7, No. 3. P. 27–33. URL: <https://doi.org/10.1109/MITP.2005.69>
17. Aberer Karl, Hauswirth Manfred, Salehi Ali. Infrastructure for Data Processing in Large-Scale Interconnected Sensor Networks. *International Conference on Mobile Data Management*. 2007. P. 198–205. URL: <https://doi.org/10.1109/MDM.2007.36>
18. M. Abbasi, A. Shahraki, J. Prieto, A.G. Arrieta, J.M. Corchado. Unleashing the Potential of Knowledge Distillation for IoT Traffic Classification. *IEEE Transactions on Machine Learning in Communications and Networking*. 2024. V. 2, P. 221-239. URL: <https://doi.org/10.1109/TMLCN.2024.3360915>
19. Loginovskii O.V., Nesterov M.I., Shestakov A.L. Application of architectural approach methods in the development of the information system of a large university. *Vestnik Yuzhno-Ural'skogo gosudarstvennogo universiteta. Seriya: Komp'yuternye tekhnologii, upravlenie, radioelektronika*. 2013. V. 13, No. 4. P. 123-128. (In Russ.)
20. Andrey Somov, Alexander Baranov, Denis Spirjakin. A wireless sensor–actuator system for hazardous gases detection and control. *Sensors and Actuators A: Physical*. 2014. V. 210, P. 157-164. URL: <https://doi.org/10.1016/j.sna.2014.02.025>
21. Filonov O.M., Bestugin A.R., Kirshina I.A., Ovchinnikova N.A., Okin P.A. Acoustic laser module for detecting small-sized unmanned aerial vehicles in an urban environment *Datchiki i sistemy*. 2023. No. 1 (266). P. 40-45. (In Russ.). URL: <https://doi.org/10.25728/datsys.2023.1.6>
22. *Xively, Platform for Internet of Things*kl. URL: <https://xively.com/>
23. *Drools, business logic integration platform*. URL: <http://www.drools.org/>

24. P. Kotak, H. Modi. Enhancing the Data Mining Tool WEKA. *2020 5th International Conference on Computing, Communication and Security (ICCCS)*, Patna, India, 2020. P. 1-6.

Статья поступила в редакцию 17.01.2025

Одобрена после рецензирования 24.01.2025

Принята к публикации 25.06.2025

The article was submitted on 17.01.2025; approved after reviewing on 24.01.2025; accepted for publication on 25.06.2025