

## 1.2. БАЗА ДАННЫХ ДЛЯ ДЕЦЕНТРАЛИЗОВАННОЙ СЕТИ ОБОРУДОВАНИЯ ИНТЕЛЛЕКТУАЛЬНОГО ЗДАНИЯ

Душкин Р. В., Москва

*В статье представлено описание распределённой базы данных, предназначенной для описания различных режимов функционирования информационных и автоматизированных систем в составе интеллектуальных зданий, а также их компонентов — в основном, различных типов периферийного оборудования (сенсоров, исполнительных устройств). Кратко описывается структура базы данных, её предназначение и цель создания. Дается видение авторов относительно вариантов использования распределённой базы данных. Новизна работы основана на применении технологий распределённого хранения информации в вопросах автоматизации интеллектуальных зданий. Актуальность работы следует из современных вызовов урбанизированному обществу по организации для людей среды с устойчивым развитием и возможностью повышения качества жизни в различных аспектах. Статья будет интересна учёным, инженерам и специалистам, работающим в области автоматизации зданий и сооружений и, более общо, автоматизации городского и муниципального управления.*

### 1. Введение

При организации распределённой вычислительной системы, обеспечивающей вычислениями интеллектуализированную систему управления зданием, важным элементом является база данных для описания режимов функционирования элементов автоматизированных и инженерных систем интеллектуального здания [Душкин & Андронов, 2019]. Эта база данных представляет собой набор вспомогательной информации о том, как функционируют и где располагаются все элементы автоматизированных и инженерных систем, входящих в реактивный контур управления [Душкин & Андронов, 2019a]. Эта информация как раз и требуется для организации распределённой вычислительной системы.

Описываемая в настоящей работе база данных представляет собой децентрализованное хранилище, разнесённое по всему периферийному оборудованию, которое может хранить информацию и осуществлять вычисления. Однако у этой базы данных имеется существенный централизованный компонент, который выполняет координирующую и диспетчеризирующую роли. Централизованный компонент при этом может представлять собой единый центр принятия решений или же центр, разнесённый на несколько объектовых комплексов для организации отказоустойчивой или катастрофоустойчивой схемы управления [Surianarayanan & Chelliah, 2019].

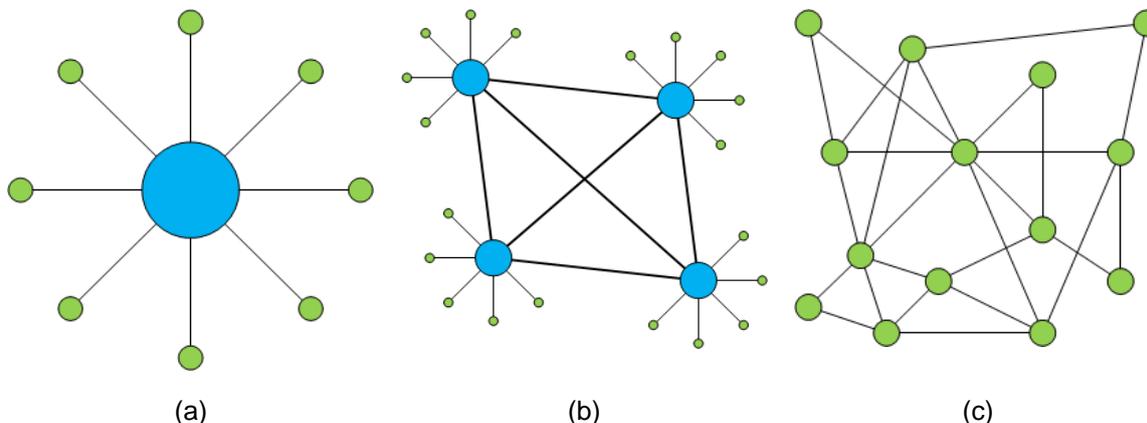


Рис 1. Модели организации вычислительных мощностей. (а) Централизованная модель. (б) Кластерная модель. (с) Распределённая модель. Голубым цветом на диаграмме обозначено серверное оборудование, зелёным — терминалы и периферийное оборудование. Линиями показаны связи между узлами в сети.

На рис. 1 представлены три модели построения вычислительной системы. Для построения базы данных и для описания режимов функционирования элементов автоматизированных и инженерных систем интеллектуального здания также подходят подобные модели. Из представленных на указанном рисунке моделей необходимо использовать централизованную и кластерную модели, только их необходимо соединить вместе для создания отказоустойчивой или катастрофоустойчивой схемы управления. Результат представлен на рис. 2.

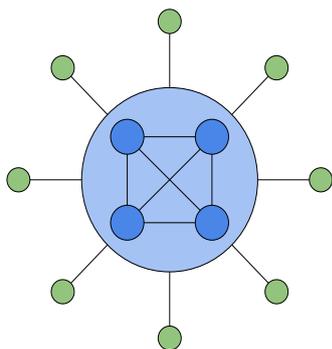


Рис. 2. Централизованно-кластерная модель организации базы данных. Голубым цветом на диаграмме обозначено серверное оборудование, зелёным — терминалы и периферийное оборудование. Линиями показаны связи между узлами в сети.

Здесь на диаграмме большим синим кругом обозначен кластер серверного оборудования, который выступает центром при организации базы данных и выглядит единым при взгляде со стороны периферийного оборудования, однако внутри этот кластер состоит из нескольких серверов, которые взаимно реплицируют информацию друг на друга для обеспечения отказоустойчивой или катастрофоустойчивой схемы хранения данных [Fujimoto, 2016]. Тем самым обеспечивается распределённая централизованность управления и распространения информации по периферийному оборудованию.

### 1. Структура базы данных

Описание режимов функционирования элементов автоматизированных и инженерных систем интеллектуального здания состоит из следующих обязательных компонентов:

1. Таблица физических и (или) логических адресов периферийного оборудования, входящего в контур реактивного управления интеллектуальным зданием и его внутренней средой. Эта таблица содержит отображение идентификаторов устройств, которые могут входить в распределённую вычислительную систему, на адреса в локальной вычислительной сети, по которым эти устройства можно вызвать. Кроме того, таблица содержит дополнительную служебную информацию о том, какие вычислительные ресурсы может предоставить соответствующее устройство для общих целей управления.
2. Таблица ссылок на функции, используемые для вычисления параметров, используемых при мониторинге и управлении параметрами внутренней среды интеллектуального здания в соответствии с вычислительной моделью. Наборы функций описываются в вычислительной системе как динамические библиотеки, которые может использовать любое совместимое устройство для вызова и запуска функций с передачей им параметров.
3. Таблица сценариев управления значениями параметров внутренней среды интеллектуального здания в соответствии с положениями, описанными в [Душкин, 2020]. В каждом сценарии задействуются определённые функции из предыдущей таблицы для вычисления различных величин, требуемых для исполнения сценария, поэтому в описании сценариев должны быть ссылки на соответствующие описания функций.

Любое периферийное устройство (сенсор или актуатор) может использоваться в качестве вычислительного узла в распределённой сети вычислений, необходимых для принятия решений, если такое периферийное устройство обладает достаточной для этого вычислительной мощностью, которая простаивает некоторое время и не используется для непосредственной работы этого устройства. В этом случае неиспользуемые вычислительные ресурсы устройства могут быть переданы в распределённую вычислительную систему для осуществления расчётов, необходимых центральной системе управления для принятия решений по воздействиям на внутреннюю среду интеллектуального здания [Душкин & Родионов, 2019].

Кроме того, все функции, которые используются для обработки информации в интеллектуализированной системе управления зданием, должны быть собраны в одну или несколько библиотек, которые доступны всем периферийным устройствам, предоставляющим свои вычислительные мощности в общую распределённую вычислительную сеть. Это значит, что в базе данных для описания режимов функционирования элементов автоматизированных и инженерных систем интеллектуального здания должны быть собраны ссылки на библиотеки с описанием сигнатур функций. В свою очередь, сигнатура функции представляет собой следующий объект:

$$f: X \rightarrow Y,$$

где:

- $f$  — идентификатор функции, по которому узлы распределённой вычислительной сети могут обратиться к ней.
- $X$  — упорядоченное множество типов аргументов функции.
- $Y$  — упорядоченное множество типов результатов вычисления функции.

Тем самым реализуется функциональный подход, детально описанный в [Душкин, 2018].

При реализации двухуровневой интеллектуальной схемы управления (с реактивной и проактивной подсистемами управления [Душкин & Андронов, 2019а]) зданием и его внутренней средой, указанные выше таблицы базы данных, которая используется для описания режимов функционирования элементов автоматизированных и инженерных систем интеллектуального здания, должны быть исчерпывающе заполнены. На основании этих таблиц осуществляется функционирование интеллектуального здания,

которое в непрерывном режиме осуществляет мониторинг и управление значениями параметров своей внутренней среды [Душкин & Родионов, 2019].

Общая логическая структура описываемой базы данных выглядит так, как показано на следующем рисунке.

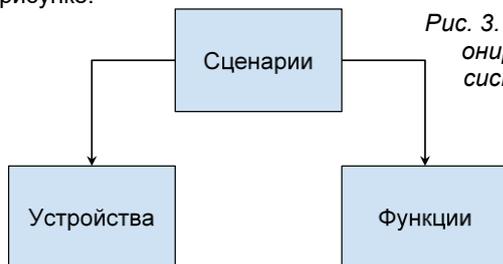


Рис. 3. Общая схема базы данных для описания режимов функционирования элементов автоматизированных и инженерных систем интеллектуального здания

Таблица сценариев содержит записи, которые ссылаются как на записи из таблицы устройств, так и на записи из таблицы функций. При этом в общем случае таблицы устройств и функций могут быть изолированными, хотя для определённых инженерных задач их также можно связывать друг с другом. Также необходимо отметить,

что на представленном рисунке дана исключительно логическая концептуальная схема базы данных, а не её физическая структура. Последняя должна быть формализована на этапе проектирования конкретной системы управления конкретного интеллектуального здания.

## 2. Варианты использования базы данных

При добавлении в состав какой-либо инженерной системы интеллектуального здания нового периферийного устройства, которое может делиться своими вычислительными мощностями, это новое устройство должно быть прописано в соответствующей таблице. Если же в контур управления интеллектуальным зданием добавляется новая инженерная система с набором периферийного оборудования, то всё множество устройств также должно быть прописано в таблицах базы данных. При выводе устройства или системы из эксплуатации для сохранения целостности распределённой вычислительной сети соответствующие устройства должны быть помечены в таблице как нефункционирующие. В этом случае сеть перестанет передавать им смарт-контракты на вычисление.

К периферийным устройствам, которые могут передавать в распределённую вычислительную сеть часть своих вычислительных мощностей, также предъявляются специфические требования. Такие устройства должны иметь возможность устанавливать дополнительное встраиваемое программное обеспечение, которое отвечает за включение устройства в распределённую вычислительную сеть и использование вычислительных ресурсов для общих целей системы управления интеллектуальным зданием. Такое встраиваемое программное обеспечение как бы является драйвером для распределённой вычислительной сети, что позволяет периферийному устройству включаться в её работу и выполнять вычисления и функции, первоначально устройству несвойственные.

Этот процесс осуществляется следующим образом:

1. В процессе *пуско-наладки* системы:
  - Каждое периферийное устройство, которое может делиться своими вычислительными мощностями, необходимо внести в соответствующую таблицу базы данных с указанием всех необходимых атрибутов.
  - На каждом таком устройстве необходимо развернуть дополнительное встраиваемое программное обеспечение, которому предоставить все необходимые права доступа для полноценной работы в качестве драйвера распределённой вычислительной сети.
  - Каждое устройство с установленным дополнительным встраиваемым программным обеспечением необходимо подвергнуть тестированию на предмет корректности работы ПО и устройства с ним в целом.
  - Наконец, вся развёрнутая вычислительная сеть также должна быть подвергнута тестированию для отработки различных сценариев взаимодействия оборудования в этой сети. Для этих целей предварительно должна быть разработана Программа и методика испытаний распределённой вычислительной сети интеллектуального здания.
2. В процессе *функционирования* системы:
  - Периферийное устройство может работать в режиме запроса. Если у него есть потребность в вычислении какого-либо набора значений, а количества вычислительных мощностей для этого не хватает, то это устройство может отправить в распределённую вычислительную сеть смарт-контракт на вычисление этих значений.
  - Периферийное устройство может работать в режиме передачи своих вычислительных мощностей для использования в распределённой вычислительной сети в случаях, когда оно или простаивает, или его мощности достаточны для обработки собственной функциональности устройства, и ещё остаются избытки. В этом случае устройство принимает смарт-контракты из направленного ациклического графа, выполняет их и записывает результаты в новые транзакции этого графа.

- Периферийное устройство также может работать в обоих режимах — как запроса на вычисления, так и вычисления чужих смарт-контрактов. Это может происходить, когда своих вычислительных мощностей не хватает на обеспечение собственной функциональности, но если их освободить от этой нагрузки, они могут быть использованы для чего-то менее затратного.
- Центральная система управления (как реактивная, так и проактивная её подсистемы) также могут участвовать в исполнении смарт-контрактов, которые появляются в направленном ациклическом графе.

Переход на этот процесс позволяет развернуть распределённую вычислительную сеть для обеспечения вычислительными потребностями системы управления и периферийного оборудования интеллектуального здания. Ожидается, что эта сеть позволит повысить эффективность работы системы управления и высвободит дополнительные вычислительные мощности для задач моделирования, прогнозирования и планирования управляющих воздействий со стороны проактивной подсистемы управления на реактивный контур управления и инженерные системы здания так, как это описано в [Душкин, 2020].

Важным моментом в работе такой распределённой вычислительной сети является необходимость обеспечения возможности кэширования исполнительных модулей, позволяющих вычислять функции, на периферийных устройствах. Эта техника позволит осуществить дополнительную экономию времени и вычислительных ресурсов, так как исключит постоянную загрузку исполняемого кода функций на периферийное оборудование, снизит нагрузку на сеть передачи данных, а также снизит затраты времени на приём-передачу информации. Даже несмотря на то, что обычно внутренняя локальная вычислительная сеть здания как транспортная среда функционирует с минимальными показателями латентности, постоянная передача кода функций между центральной базой данных и устройствами снизит эффективность общей работы распределённой вычислительной сети.

Кэширование осуществляется при помощи загрузки на периферийное устройство исполняемого кода функции для расчёта того или иного показателя, необходимого в работе интеллектуального здания. Это значит, что на периферийных устройствах, которые могут участвовать в работе распределённой вычислительной сети, желателен наличие локального хранилища данных. Полученные в локальные хранилища исполняемые коды функций для вычислений остаются в кэше до тех пор, пока эта функция требуется в работе конкретного периферийного устройства. С другой стороны, если конкретное периферийное устройство закачало к себе в кэш определённый набор функций, то в дальнейшей работе оно будет принимать на выполнение только такие смарт-контракты, которые предполагают использование только имеющихся функций.

Другими словами, встраиваемое программное обеспечение, которое должно устанавливаться на периферийное оборудование, должно обладать следующими характеристиками и свойствами:

1. Иметь многоплатформенное исполнение для того, чтобы иметь возможность быть развёрнутым на широкой номенклатуре периферийного оборудования.
2. Иметь возможность управления внутренней памятью долговременного хранения периферийного устройства для кэширования записываемых на него исполняемых кодов функций.
3. Иметь возможность балансировки запросов функций так, чтобы минимизировать перекачиваемый по локальной сети передачи данных объём исполняемого кода между центральным кластером баз данных и периферийным оборудованием инженерных систем интеллектуального здания.
4. Обеспечивать закачивание исполняемых кодов функций из базы данных на периферийное устройство с расчётом оценки будущей потребности в этой функции на конкретном устройстве и планированием её использования для исполнения смарт-контрактов в транзакциях вычислительной сети.
5. Обеспечивать выбор из направленного ациклического графа тех транзакций, в смарт-контрактах которых есть только функции, которые либо записаны в локальную память периферийного устройства, либо могут быть записаны, и при этом вычисление которых может быть осуществлено на устройстве за приемлемое время, при этом учитывать количество «токенов», которое сможет «заработать» устройство после исполнения смарт-контракта.
6. Отправлять в направленный ациклический граф смарт-контракты, в которых описаны потребности периферийного устройства в вычислении каких-либо параметров, а также получать оттуда вычисленные результаты.
7. Уметь рассчитывать текущее количество «токенов», которые находятся в распоряжении периферийного оборудования, для балансировки запросов на исполнение смарт-контрактов и осуществления вычислений смарт-контрактов других устройств.

Важным нюансом работы встраиваемого программного обеспечения является то, что оно, фактически, представляет собой «кошелёк» для содержания «токенов», находящихся в наличии у конкретного устройства. Токены, которые используются для распределения между периферийными устройствами, входящими в состав распределённой вычислительной сети, предназначены для осуществления механизмов саморегуляции этой сети. В частности, высоконагруженные устройства, которые выполняют критические для всей системы управления функции, будут иметь возможность заказывать исполнение своих смарт-контрактов за более «дорогую цену», что будет способствовать более быстрому и качественному исполнению таких заказов.

Экономическая составляющая работы распределённой вычислительной сети, которая формируется на основе описываемой базы данных для описания режимов функционирования элементов автоматизированных и инженерных систем интеллектуального здания, позволяет, кроме всего прочего, оперативно выявлять узкие места в этой распределённой вычислительной сети и принимать решения в части их усиления или исключения из сети. Например, по мере работы и перехода всей системы к состоянию равновесия во всём множестве узлов будут находиться «нищие», «средний класс» и «богачи».

К первому множеству «нищих» будут относиться узлы сети (периферийные устройства), у которых много запросов на исполнение смарт-контрактов, но которые сами их не исполняют в силу различных причин. Множество «среднего класса» будут составлять периферийные устройства, которые сбалансированы по запросам и исполнению смарт-контрактов. Наконец, к множеству «богачей» перейдут устройства, у которых много свободных вычислительных мощностей и большой объём локальной памяти, поэтому они могут всё это предоставлять для исполнения смарт-контрактов.

В следующей таблице представлено краткое описание различных экономических типов периферийного оборудования с точки зрения критичности для интеллектуальной системы управления зданиями и сооружениями.

**Таблица 1. Описание классов периферийного оборудования с точки зрения экономики и критичности для интеллектуальной системы управления**

	Критично	Некритично
<b>Нищий</b>	В этом сегменте находятся узлы (периферийное оборудование), которые выполняют критичные функции для системы управления, но которым не хватает собственных вычислительных мощностей для выполнения каких бы то ни было вычислений, поэтому им приходится тратить свои токены на заказ исполнения смарт-контрактов. В итоге токенов становится всё меньше и меньше, и центральной системе управления приходится постоянно поддерживать этот узел «дотациями». Из всех это самая негативная ситуация, которую необходимо решать внесистемными методами — например, при помощи наращивания вычислительных мощностей или замены периферийного оборудования.	Некритичные «нищие» узлы (периферийное оборудование), фактически, забирают у распределённой сети вычислительные ресурсы в большом количестве, запуская в неё смарт-контракты для вычисления требуемых значений, при этом не обладая собственными мощностями. Эта ситуация позволяет сказать, что такой тип периферийного оборудования, скорее всего, должен быть проанализирован на предмет эффективности своей работы — для чего в контуре инженерных систем интеллектуального здания функционирует устройство, которое потребляет большое количество вычислительных ресурсов, но при этом не выполняет критичной роли. Скорее всего, для повышения общей эффективности работы системы управления от таких устройств необходимо избавляться.
<b>Средний класс</b>	Средний класс характеризуется тем, что периферийные устройства, входящие в него, балансируют свои вычислительные мощности таким образом, что общее количество токенов в их распоряжении находится на среднем уровне. Это значит, что либо периферийное устройство самостоятельно справляется со своими вычислениями, либо оно умело отвечает на запросы смарт-контрактов в сети так, что расходы токенов на потребные ему высоконагруженные вычисления компенсируются исполнением смарт-контрактов, для которых требуется меньшее количество вычислительных ресурсов. В этом классе особо нет разницы, является ли устройство критичным для поддержания инженерной инфраструктуры интеллектуального здания или нет, так как в любом случае оно полностью выполняет своё предназначение. Менять что-либо во множестве устройств «среднего класса», скорее всего, не требуется, и вся система должна стремиться к тому, чтобы все её устройства принадлежали к этому классу.	
<b>Богач</b>	Класс «богачей» содержит периферийные устройства, которые зарабатывают больше токенов, чем тратят, поэтому на их кошельках осуществляется постоянное накопление. Если «богач» относится к критическим устройствам или системам, то такая ситуация, в общем-то, не может рассматриваться как негативная — устройство выполняет свою функцию и при этом также успевает сдавать в аренду свои вычислительные мощности так, чтобы пополнять свой кошелёк. Единственный вопрос, который необходимо решать в этом случае, заключается в том, чтобы устройство не занималось вычислениями некритического сегмента, а резервировало свои вычислительные мощности для повышения собственной отказоустойчивости. Устройства этого класса должны сосредотачиваться только на смарт-контрактах «нищих» устройств критической инфраструктуры.	Поскольку «богач» может сдавать свои вычислительные мощности для высоконагруженных вычислений и делает это в намного большем объёме, нежели заказывает вычисления сам, то неясна причина, по которой в класс богачей должны попадать устройства, не входящие в критический контур. Единственная рациональная причина заключается в размещении на периферии специальных вспомогательных устройств для облегчения вычислений, отсутствие которых, однако, не вызовет коллапса. Их наличие диктуется всего лишь облегчением работы. Однако в любом случае наличие подобных узлов в распределённой вычислительной сети должно наводить на размышления о наличии важных причин для присутствия таких устройств в сети — заменяя их на менее мощные, можно оптимизировать эксплуатационные расходы.

Со средним классом есть нюанс, о котором также необходимо упомянуть. Вхождение узла в множество среднего класса может свидетельствовать о двух способах экономического поведения в сети:

1. Узел закрывает только собственные потребности в вычислениях, не тратя свои токены на заказ исполнения смарт-контрактов. Это значит, что его токены просто не расходуются.
2. Узел заказывает примерно такое же количество смарт-контрактов, которое исполняет сам, так что баланс количества токенов в его кошельке меняется со временем очень незначительно, колеблясь вокруг некоторого среднего эквilibriumа.

С функциональной точки зрения различий в окончательном результате поведения таких узлов в распределённой вычислительной сети нет, однако экономическая точка зрения может вывести из соотношения количеств таких устройств в сети какие-либо выводы и рекомендации. Однако рассмотрение экономической составляющей поведения всей вычислительной сети, как уже было сказано, выходит за рамки настоящей работы.

Осталось упомянуть, что при введении экономической составляющей во взаимодействие узлов в распределённой вычислительной сети необходимо обеспечить наличие некоторого механизма эмиссии токенов для того, чтобы такой механизм распределял новые объёмы токенов по узлам сети и обеспечивал экономическое движение. Такой механизм может быть разработан на нескольких различных принципах:

1. Математический алгоритм, который начисляет некоторое количество новых токенов за подтверждение транзакции в дополнение к заработку за исполнение смарт-контрактов этой транзакции. Эти новые токены берутся как бы «из ниоткуда» и представляют собой эмиссию новых токенов для вознаграждения работающих узлов сети. Сходный механизм реализован, например, в такой блокчейн-сети, как Ethereum. Однако предполагается, что в этом методе есть проблема, которая может быть выражена следующим образом: «богатые станут ещё богаче, а бедные ещё беднее».
2. Централизованная эмиссия токенов каким-либо выделенным узлом с последующим распределением целевым образом для балансировки общей массы токенов по узлам. Этот механизм позволяет сгладить появление нищих узлов, которые больше не смогли бы заказать для себя исполнения своих смарт-контрактов без централизованных дотаций.
3. Некоторое совмещение двух предыдущих механизмов, при котором эмиссия новых токенов осуществляется децентрализованно математическим алгоритмом в каждой подтверждённой транзакции, но при этом также имеется некоторая алгоритмическая процедура, при помощи которой «богатые помогают бедным», выравнивая уровень сбережений в каждом узле сети.

### **Заключение**

Использование экономических механизмов при организации распределённой вычислительной сети на периферийных устройствах инженерных систем интеллектуального здания позволяет применить к вопросам интеллектуализации системы управления достижения теории многоагентных систем. Экономическая составляющая на направленном ациклическом графе переводит узлы вычислительной сети в ранг рациональных экономических агентов, совместное поведение которых интересно для изучения в том числе и потому, что в таком совместном поведении могут возникать эмерджентные эффекты, в том числе проявляться и «роевой интеллект» [Wanka, 2019].

Таким образом, база данных для описания режимов функционирования элементов автоматизированных и инженерных систем интеллектуального здания становится основой для разработки и запуска модели распределённых вычислений для организации программной среды, обеспечивающей управление автоматизированных и инженерных систем интеллектуального здания. Над этой моделью, основанной на использовании направленного ациклического графа (концепция, расширяющая классический блокчейн), можно организовывать различные дополнительные надстройки, например, экономику взаимодействия рациональных агентов, которыми становятся узлы вычислительной сети.

Всё перечисленное может помочь перевести процессы управления зданиями и сооружениями на новый качественный уровень, что, в свою очередь, позволит перевести их функционирование на интеллектуализированный фундамент с использованием функционального подхода [Душкин, 2018a].

### **Литература:**

1. Fujimoto K. (2016) Disaster Recovery // In book: Encyclopedia of Database Systems, January 2016. — DOI: 10.1007/978-1-4899-7993-3\_1338-2.
2. Surianarayanan C., Chelliah P. R. (2019) Disaster Recovery // In: Essentials of Cloud Computing, August 2019. — DOI: 10.1007/978-3-030-13134-0\_12.
3. Wanka R. (2019) Swarm intelligence // IT — Information Technology.— № 61, 2019. — DOI: 10.1515/itit-2019-0034.
4. Душкин Р. В. (2018) Функциональный подход в управлении технологическими процессами интеллектуальных зданий // Научно-практический журнал «Экономика и управление инновациями». — 2018. — № 4. — С. 27-35. — ISSN 2587-5574. — DOI: 10.26730/2587-5574-2018-4-27-35.
5. Душкин Р. В. (2018a) Особенности функционального подхода в управлении внутренней средой интеллектуальных зданий // Прикладная информатика, Том 13, № 6 (78), 2018. — с. 20-31. — ISSN 1993-8314.

6. Душкин Р. В., Андронов М. Г. (2019) Термин «функциональный дизайн» в отношении структурирования среды интеллектуального здания и его математическое описание в контексте повышения конкурентоспособности строительной продукции // Экономика и управление инновациями. — 2019. — № 3 (10). — С. 69-77. — DOI: 10.26730/2587-5574-2019-3-69-77.

#### References in Cyrillics

1. Fujimoto K. (2016) Disaster Recovery // In book: Encyclopedia of Database Systems, January 2016. — DOI: 10.1007/978-1-4899-7993-3\_1338-2.
2. Surianarayanan C., Chelliah P. R. (2019) Disaster Recovery // In: Essentials of Cloud Computing, August 2019. — DOI: 10.1007/978-3-030-13134-0\_12.
3. Wanka R. (2019) Swarm intelligence // IT — Information Technology. — № 61, 2019. — DOI: 10.1515/itit-2019-0034.
4. Dushkin R. V. (2018) Funkcional`ny`j podxod v upravlenii texnologicheskimi processami intellektual`ny`x zdaniij // Nauchno-prakticheskij zhurnal «E`konomika i upravlenie innova-ciyami». — 2018. — № 4. — С. 27-35. — ISSN 2587-5574. — DOI: 10.26730/2587-5574-2018-4-27-35.
5. Dushkin R. V. (2018a) Osobennosti funkcional`nogo podxoda v upravlenii vnutrennej sre-doj intellektual`ny`x zdaniij // Prikladnaya informatika, Tom 13, № 6 (78), 2018. — s. 20-31. — ISSN 1993-8314.
6. Dushkin R. V., Andronov M. G. (2019) Termin «funkcional`ny`j dizajn» v otnoshenii struk-turirovaniya sredy` intellektual`nogo zdaniya i ego matematicheskoe opisaniye v kontekste povy`sheniya konkurentosposobnosti stroitel`noj produkcii // E`konomika i upravlenie in-novatsiyami. — 2019. — № 3 (10). — С. 69-77. — DOI: 10.26730/2587-5574-2019-3-69-77.

*Душкин Роман Викторович Агентство Искусственного Интеллекта, Директор по науке и технологиям ([roman.dushkin@gmail.com](mailto:roman.dushkin@gmail.com))*

#### Ключевые слова

интеллектуальное здание, интеллектуализация, база данных, распределённая база данных, умный дом, умный город, управление, автоматизация.

#### **Roman Dushkin, Database for a decentralized network of smart building equipment**

#### Keywords

intellectual building, intellectualization, database, distributed database, smart building, smart city, management, automation

**DOI:** 10.34706/DE-2020-02-02

**JEL classification:** O33 Technological Change: Choices and Consequences • Diffusion Processes •

#### Abstract

The article presents a description of a distributed database designed to describe the various modes of operation of information and automated systems in the composition of intelligent buildings, as well as their components — mainly various types of peripheral equipment (sensors, actuators). It briefly describes the structure of the database, its purpose, and the purpose of its creation. The authors' vision of use cases of a distributed database is given. The novelty of the work is based on the use of distributed information storage technologies in the automation of intelligent buildings. The relevance of the work follows from the contemporary challenges of an urbanized society in organizing for people an environment with sustainable development and the possibility of improving the quality of life in various aspects. The article will be of interest to scientists, engineers, and specialists working in the field of building and structure automation and, more generally, automation of city and municipal government.