

На правах рукописи



СИНЮКОВ Денис Сергеевич

**СПЕЦИАЛЬНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ПРОЦЕССА
УПРАВЛЕНИЯ ТРАНЗАКЦИЯМИ С ОПЕРАТИВНЫМ КОНТЕНТОМ
НА ОСНОВЕ РАСПРЕДЕЛЕННОГО КЭШИРОВАНИЯ**

Специальность: 2.3.5. Математическое и программное обеспечение
вычислительных систем, комплексов и
компьютерных сетей

АВТОРЕФЕРАТ

диссертации на соискание ученой степени
кандидата технических наук

Воронеж – 2023

Работа выполнена в ФГБОУ ВО «Воронежский государственный технический университет».

Научный руководитель: **Данилов Александр Дмитриевич**, доктор технических наук, профессор

Официальные оппоненты: **Мельник Эдуард Всеволодович**, доктор технических наук, ФГБУН «Федеральный исследовательский центр Южный научный центр Российской академии наук», главный научный сотрудник, заведующий лабораторией информационных технологий и процессов управления (г. Ростов-на-Дону)

Лавлинский Валерий Викторович, доктор технических наук, доцент, ФГКВООУ ВО «Военный учебно-научный центр Военно-воздушных сил «Военно-воздушная академия имени профессора Н.Е. Жуковского и Ю.А. Гагарина», старший научный сотрудник научно-исследовательского центра образовательных и информационных технологий (г. Воронеж)

Ведущая организация: Федеральное государственное бюджетное образовательное учреждение высшего образования **"Воронежский государственный университет инженерных технологий" (ФГБОУ ВО "ВГУИТ")** (г. Воронеж)

Защита состоится «14» апреля 2023 года в 12:00 часов в конференц-зале на заседании диссертационного совета 24.2.286.04, созданного на базе ФГБОУ ВО «Воронежский государственный технический университет», по адресу: г. Воронеж, Московский просп., д. 14, ауд.216.

С диссертацией можно ознакомиться в научно-технической библиотеке ФГБОУ ВО «Воронежский государственный технический университет» и на сайте www.cchgeu.ru.

Автореферат разослан « 20 » февраля 2023 г.

Ученый секретарь
диссертационного совета

Гусев Константин Юрьевич

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы. Активный рост информационных потребностей общества в специальном программном обеспечении, взаимодействующем с облачными хранилищами как СУБД, способствовал созданию множества методов его проектирования. Большой вклад в их развитие внесли Мурзин Ф.А., Сахаров Д.В., Царегородцев А.В., Dar S., Jaziri W., Stankovic J.A., Zhang Q., Zhao Z. В теоретическом плане ряд методов управления транзакциями в облачных средах сводится к оптимизации выбора направлений и локальных баз в распределенной системе внутри облака. Разумной идеей кажется создание алгоритма локального кэширования хронологически запрошенных данных транзакций, обеспечивающий уменьшение время задержки передачи данных о специальных транзакциях. Особенностью механизма распределения данных о специальных транзакциях в реальном времени должно являться использование распределенного кэша.

Несмотря на глубоко развитую теорию фиксации транзакций. В данной области ведущие исследователи Болодурина И.П., Борисенко О.Д., Никульчев Е.В., Zatwarnicki K. развили множество подходов. Ключевые показатели для этих подходов, к сожалению, не учитывают интегрированную информацию о состоянии кэширования, предполагаемом размере данных и пропускной способности. Необходима метрика, учитывающая эту информацию, для минимизации задержки передачи данных транзакции.

СУБД реального времени часто проходит через периоды перегрузки после неожиданного поступления пользовательских транзакций. В такие периоды транзакции с большей вероятностью пропускают свои предельные сроки, и это напрямую влияет на QoS (качество обслуживания), предоставляемое пользователям. Таким образом, необходимы дополнительные исследования в области развития протоколов планирования транзакций в СУБД реального времени, учитывающих не только временные ограничения транзакций, но и критерии, установленные пользователями базы данных, обеспечивающие корректное определение приоритетов выполняемых транзакций. Вытекающая отсюда проблема – необходимость создания архитектуры СУБД реального времени, отличающейся применением модифицированного протокола планирования транзакций и улучшающей качество предоставления услуг пользователям при реализации управления с обратной связью.

Таким образом, **актуальность** тематики диссертационного исследования продиктована необходимостью разработки специального программного обеспечения управления транзакциями с оперативным контентом на основе распределенного кэширования.

Тематика диссертационной работы соответствует научному направлению ФГБОУ ВО «Воронежский государственный технический университет» «Вычислительные комплексы и проблемно-ориентированные системы управления».

Целью работы является разработка средств специального программного обеспечения управления транзакциями с оперативным контентом на основе распределенного кэширования с использованием модифицированного

протокола планирования транзакций.

Задачи исследования. Для достижения поставленной цели необходимо решить следующие задачи:

1. Разработать механизм распределения данных о специальных транзакциях с оперативным контентом, обеспечивающий минимизацию времени передачи данных.

2. Разработать алгоритм локального кэширования хронологически запрошенных данных транзакций, обеспечивающий уменьшение время задержки передачи данных о специальных транзакциях

3. Разработать алгоритм разделения и перенаправления запросов между клиентами, межобъектными интерфейсами или облаком, дополнительно учитывающий интегрированную информацию о состоянии кэширования, предполагаемом размере данных и пропускной способности.

4. Модифицировать протокол планирования транзакций в СУБД реального времени для учета не только временных ограничений транзакций, но и критериев, установленных пользователями базы данных.

5. Предложить архитектуру СУБД реального времени с использованием модифицированного протокола планирования транзакций для улучшения качества предоставления услуг пользователям при реализации управления с обратной связью.

6. Создать программный комплекс для проверки динамических связей технологических схем и баз данных на основе системы кодирования Kraftwerk Kennzeichen System.

7. Создать компоненты специального программного обеспечения распределенной информационно-вычислительной системы управления с унифицированными механизмами межмодульного взаимодействия для оптимизации состава резервного оборудования и исключения программных конфликтов при обмене данными.

Объект исследования: процессы управления распределенными базами данных.

Предмет исследования: математическое и программное обеспечение управления транзакциями с оперативным контентом на основе распределенного кэширования.

Методы исследования. При решении поставленных в диссертации задач использовались методы теории вероятностей, теории принятия решений, а также методы объектно-ориентированного программирования.

Тематика работы соответствует следующим пунктам паспорта специальности 2.3.5. Математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей: п.3 «Модели, методы, архитектуры, алгоритмы, языки и программные инструменты организации взаимодействия программ и программных систем»; п.9 «Модели, методы, алгоритмы, облачные технологии и программная инфраструктура организации глобально распределенной обработки данных».

Научная новизна работы. В диссертации получены следующие результаты, характеризующиеся научной новизной:

1. Механизм распределения данных о специальных транзакциях с оперативным контентом, отличающийся распределенным кэшированием в гетерогенных межобъектных интерфейсах в реальном масштабе времени, обеспечивающий минимизацию времени передачи данных.

2. Алгоритм локального кэширования хронологически запрошенных данных транзакций, отличающийся учетом собственной емкости межобъектного интерфейса и применением политикой вытеснения давно неиспользуемых данных, обеспечивающий уменьшение время задержки передачи данных о специальных транзакциях.

3. «Жадный» алгоритм разделения и перенаправления запросов между клиентами, межобъектными интерфейсами или облаком, отличающийся интеграцией информации о состоянии кэширования, предполагаемом размере данных и пропускной способности и обеспечивающий минимизацию задержки передачи данных транзакции.

4. Модифицированный протокол планирования транзакций в СУБД реального времени, учитывающий не только временные ограничения транзакций, но и критерии, установленные пользователями базы данных, обеспечивающий корректное определение приоритетов выполняемых транзакций.

5. Архитектура СУБД реального времени, отличающаяся применением модифицированного протокола планирования транзакций и улучшающая качество предоставления услуг пользователям при реализации управления с обратной связью.

Теоретическая и практическая значимость исследования заключается в разработке программного обеспечения управления транзакциями с оперативным контентом на основе распределенного кэширования с использованием модифицированного протокола планирования транзакций, а также информационного и программного обеспечения для проверки динамических связей технологических схем и баз данных, использующего систему кодирования Kraftwerk Kennzeichen System (является *стандартом де факто* в проектировании оборудования и систем управления для энергетики, но применима к любым техническим объектам).

Положения, выносимые на защиту

1. Механизм распределения данных о специальных транзакциях с оперативным контентом, отличающийся распределенным кэшированием в гетерогенных межобъектных интерфейсах в реальном масштабе времени, обеспечивает минимизацию времени передачи данных.

2. Алгоритм локального кэширования хронологически запрошенных данных транзакций, отличающийся учетом собственной емкости межобъектного интерфейса и применением политикой вытеснения давно неиспользуемых данных, обеспечивает уменьшение время задержки передачи данных о специальных транзакциях.

3. «Жадный» алгоритм разделения и перенаправления запросов между клиентами, межобъектными интерфейсами или облаком, отличающийся интеграцией информации о состоянии кэширования, предполагаемом размере данных и пропускной способности, обеспечивает минимизацию задержки

передачи данных транзакции.

4. Модифицированный протокол планирования транзакций в СУБД реального времени, учитывающий не только временные ограничения транзакций, но и критерии, установленные пользователями базы данных, обеспечивает корректное определение приоритетов выполняемых транзакций.

5. Архитектура СУБД реального времени, отличающаяся применением модифицированного протокола планирования транзакций, улучшает качество предоставления услуг пользователям при реализации управления с обратной связью.

Результаты внедрения. Основные результаты внедрены в Филиале АО «Концерн Росэнергоатом» «Нововоронежская атомная станция» в виде специальных программно-технических компонентов в информационно-вычислительной системе 4-го энергоблока, в учебный процесс Воронежского государственного технического университета в рамках дисциплин: «Управление вычислительными системами и сетями», «Современные системы управления базами данных», а также в рамках курсового и дипломного проектирования.

Апробация работы. Основные положения диссертационной работы докладывались и обсуждались на следующих конференциях: Всероссийской научной конференции «Достижения науки и технологий-ДНиТ-2021» (Красноярск, 2021); II International Scientific Conference on Advances in Science, Engineering and Digital Education: (ASEDU-II-2021) (Krasnoyarsk, October 2021); XXVII-th International Open Science Conference «Modern informatization problems in simulation and social technologies (MIP-2022'SCT)» (Yelm, WA, USA, January 2022); XVII Междунар. конф. по электромеханике и робототехнике «Завалишинские чтения'22» (Санкт-Петербург, 2022), а также на научных семинарах кафедры Компьютерных интеллектуальных технологий проектирования (2020-2022 гг.).

Достоверность результатов обусловлена корректным использованием теоретических методов исследования и подтверждена результатами вычислительных экспериментов и внедрения.

Публикации. По результатам диссертационного исследования опубликовано 12 научных работ, в том числе 8 – в изданиях, рекомендованных ВАК РФ (из них 1 – в издании Scopus и два свидетельства о регистрации программы для ЭВМ). В работах, опубликованных в соавторстве и приведенных в конце автореферата, лично автором получены следующие результаты: [1, 10] - механизм распределения данных о специальных транзакциях с оперативным контентом; [2, 6] алгоритм локального кэширования хронологически запрошенных данных транзакций, отличающийся учетом собственной емкости межобъектного интерфейса и применением политикой вытеснения давно неиспользуемых данных; [9] - «жадный» алгоритм разделения и перенаправления запросов между клиентами, межобъектными интерфейсами или облаком; [3] - модифицированный протокол планирования транзакций в СУБД реального времени, учитывающий не только временные ограничения транзакций, но и критерии, установленные пользователями базы данных; [7, 8, 12] - архи-

тектура СУБД реального времени, отличающаяся применением модифицированного протокола планирования транзакций.

Структура и объем работы. Диссертационная работа состоит из введения, четырех глав, заключения, списка литературы и приложения. Основной текст содержит 147 страниц, 31 рисунок и 9 таблиц. Список библиографических источников включает 175 наименований.

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Во введении обоснована актуальность исследования, сформулированы его цель и задачи, научная новизна и практическая значимость полученных результатов, приведены сведения об апробации и внедрении работы.

В первой главе исследуются особенности разработки математического и программного обеспечения математического и программного обеспечения управления транзакциями с оперативным контентом на основе распределенного кэширования с использованием модифицированного протокола планирования транзакций, и анализируется современное состояние проблемы управления ими. Отмечено, что повысить эффективность управления транзакциями можно путем модификации распределения данных о них в сети, создания алгоритмов и протоколов планирования транзакций с учетом дополнительных параметров, определяющих качество обслуживания. Исходя из этого, проанализированы существующие подходы к решению данных задач. Результат данного анализа потребовал формализации данных задач, а также алгоритмизации их решения с учетом особенностей, отраженных на рис. 1.

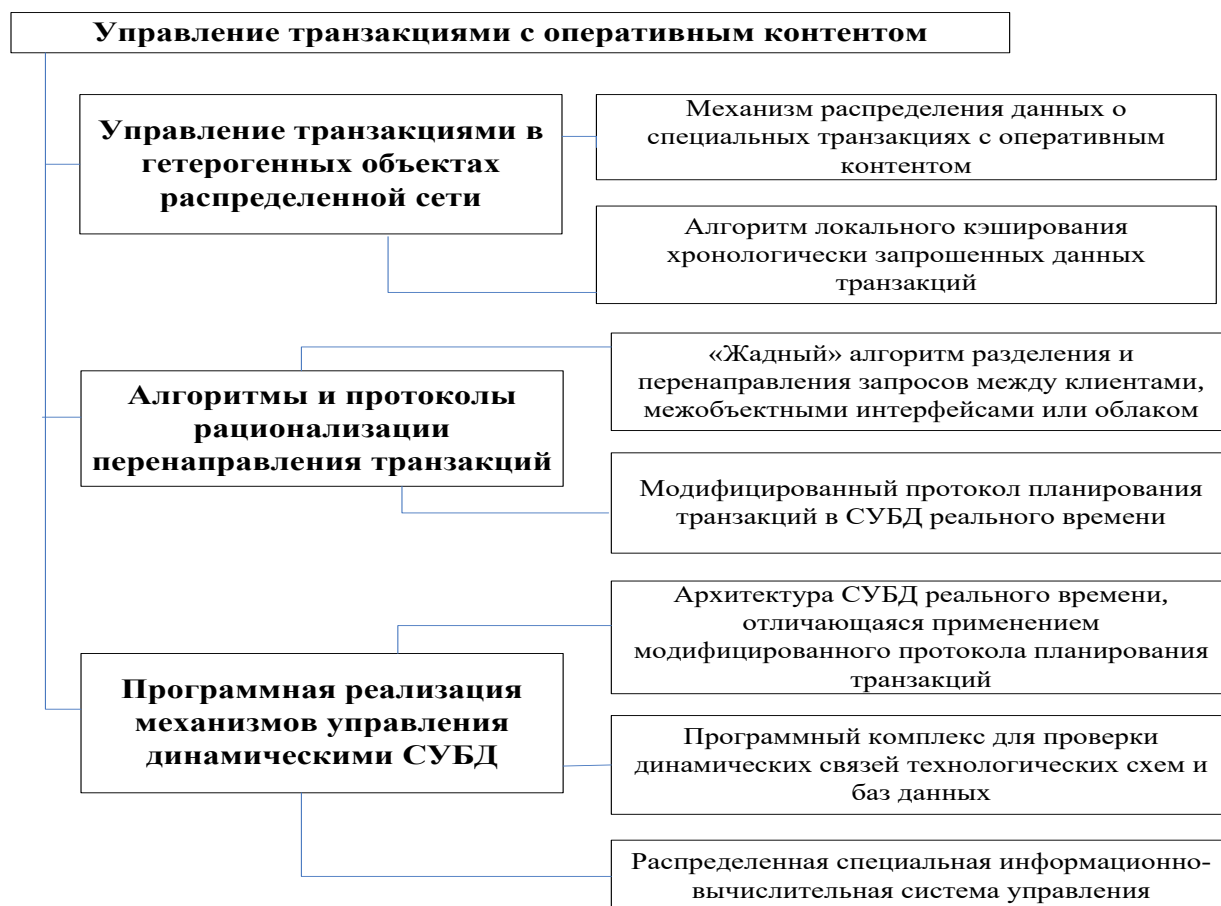


Рис. 1. Дизайн исследования

Сформулирована цель и задачи исследования, распределен материал по главам.

Вторая глава посвящена разработке новых алгоритмов управления транзакциями в гетерогенных объектах распределенной сети.

Предлагается механизм распределения данных о специальных транзакциях в реальном времени на основе распределенного кэша. Распределенный кэш в основном состоит из двух частей: одна - кэш межобъектного интерфейса (МОИН), а другая - кэш небольшой емкости конечного клиента. Работа происходит следующим образом.

1. Облако анализирует популярность различных акций в соответствии с историческими запросами “от-до” соответствующей платформы МОИН. Основываясь на вычислении возрастающей популярности и динамическом состоянии сети, облако отправляет более популярные данные о транзакциях на соответствующий межобъектный интерфейс, когда сеть имеет достаточную доступную пропускную способность.

2. Межобъектный интерфейс сообщает информацию о своих кэшированных данных другим МОИН, сохраняя при этом информацию о кэшированном содержимом локальных клиентов. Получив запрос от клиента, межобъектный интерфейс выбирает оптимальную стратегию передачи в соответствии со статусом сети и информацией о распределенном кэше. Он может перенаправить запрос в локальный кэш, другие терминалы, другие МОИН или удаленное облако. Цель состоит в том, чтобы достичь наименьшего времени завершения передачи.

3. Клиент кэширует соответствующие хронологически запрошенные данные транзакции в зависимости от своей емкости в соответствии с политикой вытеснения давно неиспользуемых данных (LRU - Least Recently Used). Для того чтобы оценить эффективность схемы, произведено сравнение время выполнения запросов на данные о специальных транзакциях.

Как показано на рис. 2, облачная платформа отвечает за хранение всей информации о транзакциях на локальном клиенте, а межобъектный интерфейс отвечает за ускорение передачи данных о транзакциях и поддержание информации распределенного кэша.

Как показано на рис. 3, после поступления запроса пользователя межобъектный интерфейс рационально разделяет соответствующий запрос контента на основе таблицы сведений о локальном и распределенном кэше и соответствующего состояния сети и перенаправляет разделенные подзапросы на другие МОИН, локальные узлы транзакций или облако. Цель принятия решения состоит в том, чтобы свести к минимуму время, необходимое клиенту транзакции для получения полных данных. При реализации, чтобы избежать затрат времени во время создания сеанса передачи, МОИН поддерживает пул соединений с облаком, который используется для немедленной передачи данных о специальных транзакциях в режиме реального времени в любое время.

Клиенты транзакций динамически запрашивают данные о транзакциях с контентом. Запросы от разных клиентов имеют соответствующую локальность, что в основном проявляется в их очевидном предпочтении знания о

конкретных видах контентов. Каждый клиент может кэшировать определенный объем контента, а емкость кэша намного меньше, чем у межобъектного интерфейса. Но общее количество клиентов значительно превышает количество межобъектных интерфейсов. Данные, запрошенные клиентом, кэшируются локально до тех пор, пока кэшированное содержимое не превысит емкость локального хранилища, и заменяются в соответствии с политикой LRU. Кроме того, как только запрос клиента будет перенаправлен другим клиентам, статус передачи между ними будет отправлен на локальную пограничную платформу в качестве основы для интеллектуального решения.

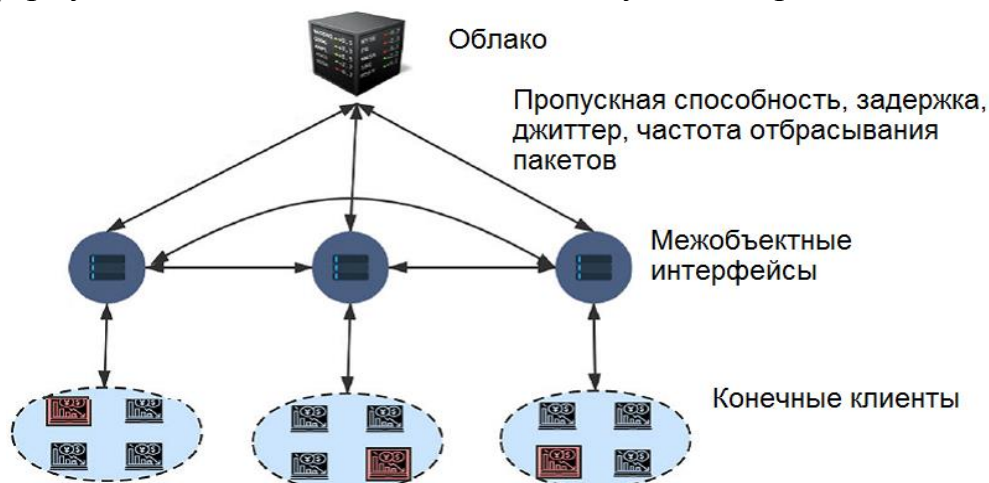


Рис. 2. Структура распределенной передачи данных в реальном времени на основе межобъектного интерфейса

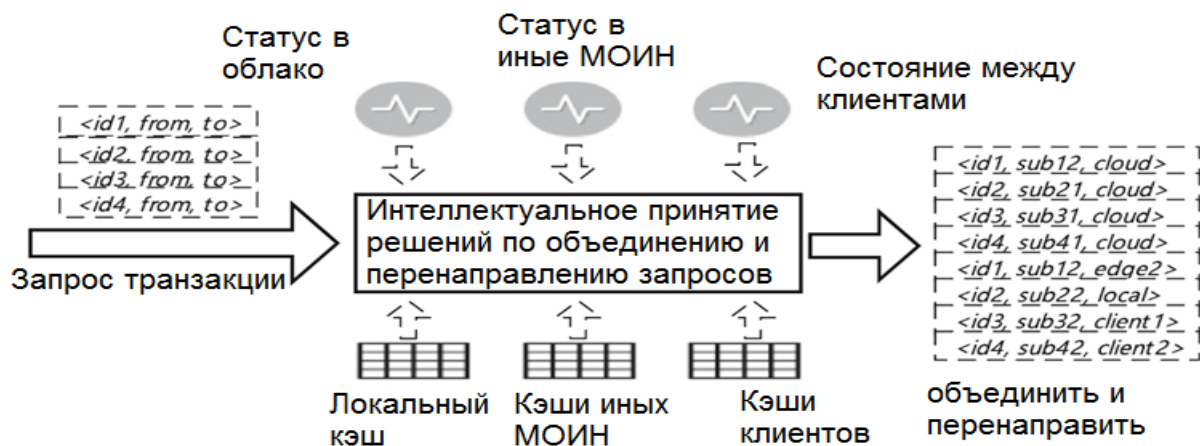


Рис. 3. Структура интеллектуального решения на основе межобъектных интерфейсов: объединение запросов и перенаправление

Таким образом, создан механизм распределения данных о специальных транзакциях с оперативным контентом, отличающийся распределенным кэшированием в гетерогенных межобъектных интерфейсах в реальном масштабе времени, обеспечивающий минимизацию времени передачи данных.

Чтобы избежать передачи большого объема данных о транзакциях при поступлении запроса информации о транзакциях с акциями, облако заранее отправляет исторические данные о транзакциях с контентом на соответ-

ствующий межобъектный интерфейс. Таким образом, когда запрос действительно выполняется, передача данных в облако может быть значительно сокращена, что обеспечивает более быстрый ответ на запрос. Однако, как ресурсы пропускной способности, так и емкость кэша МОИН ограничены, поэтому очень важно, как передать соответствующие данные о специальных транзакциях на соответствующий межобъектный интерфейс. Как правило, более важная информация о транзакциях должна передаваться на соответствующий МОИН, чтобы максимизировать эффект, обеспечиваемый возможностями пограничного кэширования.

В статистике важности учитываются только идентификаторы контентов и не проводится различий между различными временными срезами. Предполагая, что d представляет идентификатор контента, контент запрашивается в момент времени t , затем запускается обновление его важности. Предположим, что контент был запрошен в общей сложности k раз, и время запроса равно $t_1, t_2, \dots, t_k$, где $t_k = t$. После запроса в момент времени t его соответствующая важность $p(d, k)$ вычисляется так:

$$\begin{aligned} p(d, k) &= \sum_{i=1}^k e^{-\lambda(t-t_i)} = e^{-\lambda(t-t_k)} + \sum_{i=1}^{k-1} e^{-\lambda(t-t_{k-1})} e^{-\lambda(t_{k-1}-t_i)} = \\ &= e^{-\lambda(t-t_k)} + e^{-\lambda(t-t_{k-1})} \sum_{i=1}^{k-1} e^{-\lambda(t_{k-1}-t_i)} = e^{-\lambda(t-t_k)} + e^{-\lambda(t-t_{k-1})} p(d, k-1) \end{aligned} \quad (1)$$

Как показывает (1), каждый запрос вносит свой вклад в важность, но вклад запроса в важность со временем экспоненциально уменьшается. Преимущество этого расчета важности заключается в том, что каждый расчет необходимо только постепенно обновлять по предыдущему результату, что значительно снижает стоимость расчета важности.

Однако после запроса данных о контентах создается высокая важность. Но по прошествии длительного периода времени он больше не запрашивается, тогда важность контента не может быть обновлена, что приводит к большому отклонению от фактической важности. Это может снизить эффективность кэширования, поскольку некоторые неважные данные будут смещены к краям. Поэтому в предлагаемой схеме будем периодически обновлять значения важности всех контентов экспоненциальным образом на ΔT . Таким образом, значение важности контента, который не запрашивался в течение длительного времени, будет постоянно снижаться, чтобы избежать неправильной оценки важности.

В реальной сетевой среде важность рассчитывается для каждого межобъектного интерфейса, а запросы также классифицируются в соответствии с соответствующим МОИН. Как только облако обнаружит, что сеть имеет достаточную доступную пропускную способность, а межобъектный интерфейс располагает достаточным пространством кэша, облако выберет наиболее важную информацию о специальных транзакциях из списка важности, соответствующего граничным данным, и передаст ее на соответствующий межобъектный интерфейс.

Таким образом, представлен алгоритм локального кэширования хронологически запрошенных данных транзакций, отличающийся учетом собственной емкости МОИН и применением политики вытеснения давно неиспользуемых данных и обеспечивающий уменьшение время задержки передачи данных о специальных транзакциях.

Для оптимального разделения и перенаправления пользовательских запросов было разработано интеллектуальное решение на основе «жадного» алгоритма. Суть его заключается в следующем: пусть $R = \{ \langle d_1, S_1 \rangle, \langle d_2, S_2 \rangle, \dots, \langle d_k, S_k \rangle \}$ - запрос, где d_i есть идентификатор контента ID и S_i - время запроса; $L = \{ \langle Ld_i, LS_i \rangle \}$ - локальный набор данных кэширования специальных транзакций; $E_j = \{ \langle Ed_{ji}, ES_{ji} \rangle \}$ - набор данных кэширования специальных транзакций на j -м МОИН; $C_m = \{ \langle Cd_{mi}, CS_{mi} \rangle \}$ - набор данных кэширования о специальных транзакциях у m -го клиента.

Пусть $B(x, y)$ - пропускная способность между узлами x и y , где x и y могут быть клиентами, МОИН или облаком.

Далее для каждого $\langle d_i, S_i \rangle \in R$, выбираем клиента, МОИН или облако, которые могут минимизировать задержку передачи данных транзакции, на основе информации о состоянии кэширования, предполагаемом размере данных и пропускной способности $B(x, y)$. Решение о разделении и перенаправлении запросов принимается с «жадностью», чтобы свести к минимуму общее время передачи.

Таким образом, реализован «жадный» алгоритм разделения и перенаправления запросов между клиентами, межобъектными интерфейсами или облаком, отличающийся интеграцией информации о состоянии кэширования, предполагаемом размере данных и пропускной способности, и обеспечивающий минимизацию задержки передачи данных транзакции.

Третья глава посвящена разработке алгоритмов и протоколов рационализации перенаправления транзакций.

Для борьбы с фазами нестабильности в СУБД реального времени (РВ) из-за непредсказуемого поступления транзакций пользователей предложена архитектура, основанная на обратной связи, которая рассматривается как базовая архитектура для управления QoS в СУБД РВ.

В предлагаемой архитектуре рассматриваются транзакции с жестким предельным сроком, когда, если транзакция пропустит свой предельный срок, она будет прервана и станет бесполезной для системы. Будем различать два типа транзакций в реальном времени в зависимости от объекта данных, к которому осуществляется доступ: обновленные транзакции и транзакции пользователей. Обновленные транзакции - это те, которые периодически выполняются для обновления данных в реальном времени. Однако пользовательские транзакции, которые являются аperiodическими, состоят из набора операций чтения как с данными в реальном времени, так и с данными, не относящимися к реальному времени, и операций записи только с данными, не относящимися к реальному времени.

Основным критерием эффективности, рассматриваемым в предлагаемой архитектуре, является уровень удовлетворенности пользователей системы. Он представляет коэффициент успешности (SR) транзакций, которые квалифицируются как критические для пользователей. SR определяется как параметр QoS, измеряющий процент транзакций, которые соответствуют своим срокам, и имеющий следующую формулу:

$$SR = 100 \times \frac{\#Timely}{\#Late + \#Timely} (\%) \quad (2)$$

где #Late и #Timely соответственно означают количество критических транзакций, которые пропустили свои предельные сроки, и количество успешных критических транзакций.

Существует архитектура, использующая управление планированием на основе обратной связи для управления QoS в СУБД РВ. Она называется FCSA (архитектура управления планированием с обратной связью), и позволяет контролировать поведение СУБД РВ, делая его более надежным в периоды нестабильности (рис. 4).

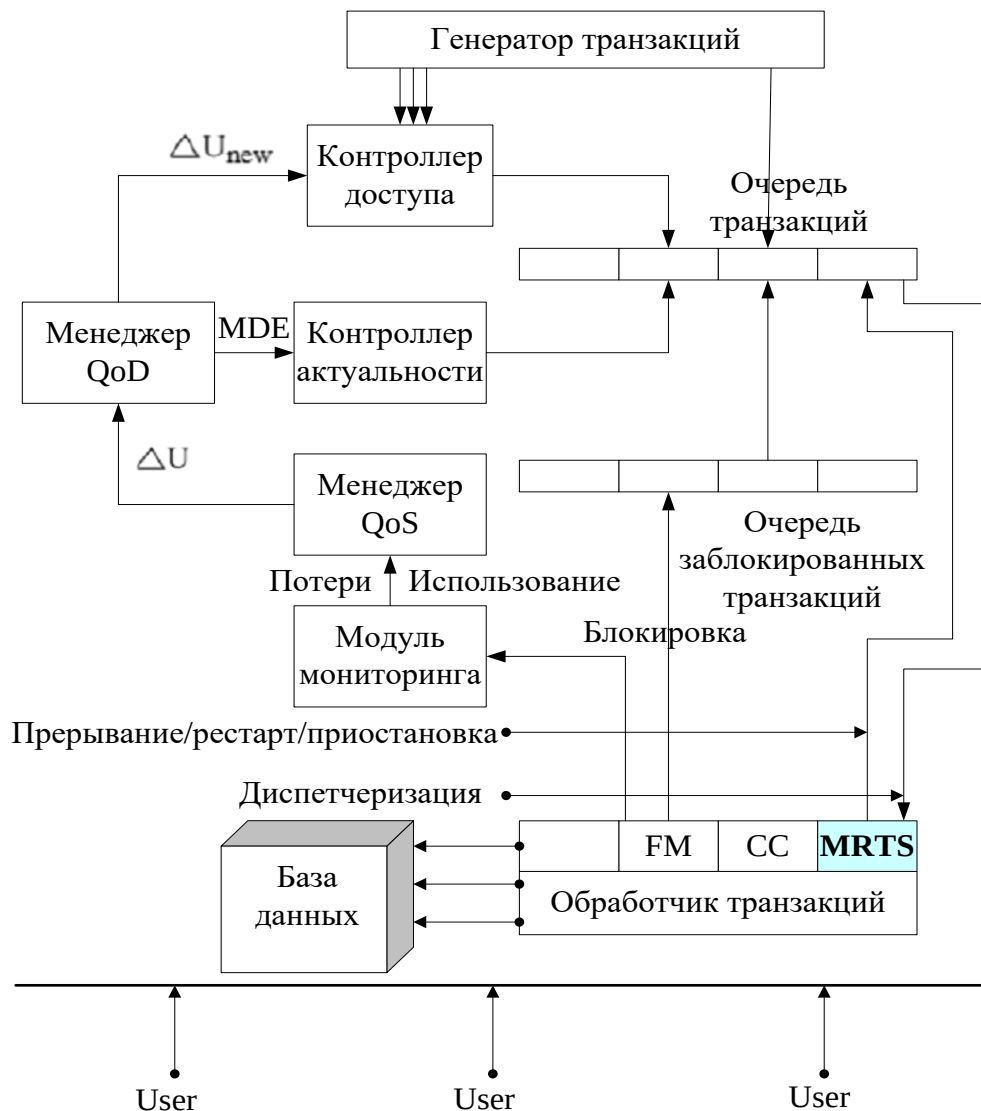


Рис. 4. Структура протокола MRTS в FCSA в составе новой архитектуры СУБД РВ

Контроллер допуска отвечает за фильтрацию транзакций пользователей. Его функционирование контролируется контуром обратной связи для управления в периоды нестабильности системы. Количество принятых транзакций зависит от состояния системы, о котором сообщает монитор, и от параметров QoS, заданных администратором базы данных (DBA).

Принятые транзакции отправляются в очередь готовности для исполнения обработчиком транзакций. Обработчик транзакций состоит из диспетчера актуальности (FM), проверяющего актуальность данных в реальном времени перед доступом к транзакциям, контроллера параллелизма (CC), разрешающего конфликты данных доступа, возникающие между транзакциями, и базового планировщика (BS), определяющего время выполнения каждой транзакции. Монитор отвечает за измерение производительности системы на основе результатов выполнения, предоставляемых обработчиком транзакций. Эти измерения относятся к контуру обратной связи и отправляются менеджеру QoS. Последний корректирует значения параметров QoS, которые будут отправлены контроллеру допуска и менеджеру QoD (качество данных). Этот параметр настроит значение максимальной ошибки данных MDE, а затем отправит его на контроллер актуальности, который используется для удаления транзакции обновления, обращающейся к данным, которые считаются устаревшими ($DF < MDE$; DE – ошибка данных).

Контур обратной связи, обеспечивающий балансировку системы, основан на принципе наблюдения и самоадаптации. Самоадаптация происходит на протяжении всего функционирования системы, чтобы постоянно корректировать ее рабочую нагрузку при наличии непредсказуемых запросов пользователей. Однако наблюдение заключается в рассмотрении состояния системы, чтобы определить, соответствует ли оно заданным параметрам QoS, таким как коэффициент использования системы и коэффициент успешности транзакций. Система адаптирует свои настройки к соответствующим изменениям в поведении и, таким образом, справится с периодами нестабильности.

Подход заключается в том, чтобы предложить модифицированный протокол планирования транзакций в СУБД РВ. Модификация заключается в учете не только временных ограничений транзакций, но и критериев, установленных пользователями базы данных. Функционирование этого протокола основано на использовании трех параметров для определения приоритетов выполняемых транзакций. Первый параметр - это предельные сроки транзакции, которые должны быть соблюдены для ее совершения. Вторым параметром является время поступления каждой транзакции в систему. Третий параметр служит показателем уровня критичности для каждой транзакции и отражает, в частности, ее важность для заинтересованного пользователя.

Поэтому считаем наиболее приоритетной транзакцию, имеющую как самый ранний предельный срок, самое раннее время прибытия, так и самый высокий уровень критичности. Таким образом, определим следующим образом интегральный приоритет каждой транзакции t_i :

$$\text{Priority}(t_i) = 1/DL(t_i) + 1/AT(t_i) + CR(t_i), \quad (3)$$

где $DL(t_i)$ представляет предельный срок транзакции t_i , $AT(t_i)$ - время ее прибытия, а $CR(t_i)$ - уровень ее критичности. Этот модифицированный протокол MRTS (смешанное планирование в реальном времени), следует принципу политики упреждающего планирования. Таким образом, транзакция с более низким приоритетом может быть прервана в пользу другой транзакции с более высоким приоритетом. Однако упреждение может быть осуществлено только после проверки того, что оно не приведет к приостановке транзакции с пропущенным крайним сроком. Это означает, что приостановленная транзакция может быть возобновлена позже с момента ее приостановки.

Итак, представлен модифицированный протокол планирования транзакций в СУБД реального времени, учитывающий не только временные ограничения транзакций, но и критерии, установленные пользователями базы данных, обеспечивающий корректное определение приоритетов выполняемых транзакций.

Протокол MRTS применен в FC-SA, который предлагается для управления QoS в СУБД РВ. Эта архитектура обеспечивает хорошую производительность для стабилизации системы в периоды перегрузки или недостаточного использования, вызванного неожиданным поступлением пользовательских транзакций. На рис. 4 показана архитектура FC-MRTS, содержащая модифицированный планировщик.

Задача предложенного подхода состоит в том, чтобы удовлетворить пользователей СУБД РВ за счет улучшения качества предоставляемых услуг, а именно за счет увеличения числа транзакций, которые соответствуют их срокам, особенно тех, которые квалифицируются как более важные для пользователей. Кроме того, совместно с FC-SA снижается риск перегрузки системы, когда одновременно происходит значительное количество транзакций.

В дальнейших исследованиях планируется ввести в подход новый параметр, отражающий связи агрегирования, которые могут существовать между транзакциями. Поэтому, если транзакция t_i является субтранзакцией t_j , необходимо начать с запуска t_i , даже если она имеет более низкий приоритет. Реализация такого решения требует анализа взаимосвязей транзакций, которые обычно недостаточно очевидны.

Чтобы оценить предложенный подход к планированию, использован симулятор, основанный на FC-SA и имитирующий поведение СУБД РВ. Проведен ряд экспериментов, в которых варьировали значения некоторых параметров. Системные параметры, используемые для генерации данных и транзакций, обобщены в табл. 1.

Системные параметры		Таблица 1
Параметр	Содержание	Значения
Duration (мс)	Длительность имитационного моделирования	10000
Validity (мс)	Время действия данных	[400, 800]
NbOfOperations	Число транзакций	[3, 10]
ReadOpTime	Время выполнения транзакции чтения	[1, 2]
WriteOpTime	Время выполнения транзакции записи	[1, 10]
MDE	Максимальное количество ошибок данных	3

Для разрешения конфликтов, возникающих между транзакциями, выбран протокол управления параллелизмом 2PL-HP (двухфазная фиксация с высоким приоритетом). Его принцип заключается в обеспечении того, чтобы выполнение транзакции с высоким приоритетом не затруднялось выполнением другой транзакции с более низким приоритетом.

В работе проводилось моделирование поведения СУБД РВ с использованием протоколов планирования EDF, FIFO и предложенного MRTS (рис. 5).

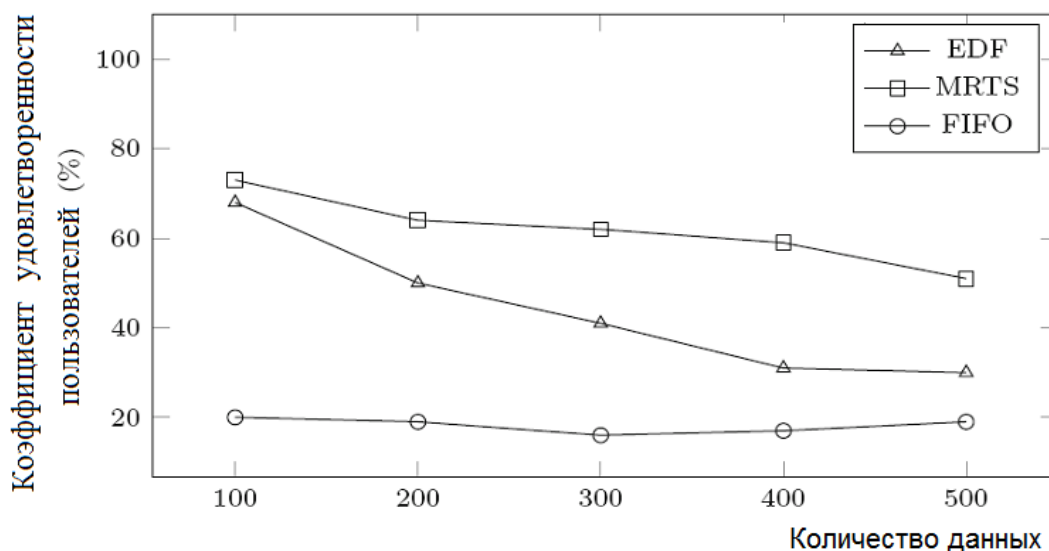


Рис. 5. Коэффициент удовлетворенности пользователей для трех планировщиков

Таким образом, разработана архитектура СУБД реального времени, отличающаяся применением модифицированного протокола планирования транзакций и улучшающая качество предоставления услуг пользователям при реализации управления с обратной связью.

В главе 4 представлены особенности программной реализации механизмов управления динамическими СУБД, основанные на результатах предшествующих глав.

Безопасность эксплуатации потенциально опасных производств является первостепенной целью в промышленности. При управлении такими объектами одной из причин возникновения нарушений являются ошибочные действия оперативного персонала. Одним из инструментов решения данной проблемы является программная проверка результатов человеческой деятельности. Для реализации такой задачи разработан программный комплекс, предназначенный для проверки динамических связей между графическими объектами интерактивных схем технологического процесса и соответствующих переменных баз данных.

В качестве методики была использована система кодирования Kraftwerk Kennzeichen System (KKS), позволяющая получать идентификаторы для типовых объектов. Результатом работы является информация о кодах KKS, которые не соответствуют базам данных. Структура ОЗУ-резидентной базы данных реального времени представлена на рис. 6.

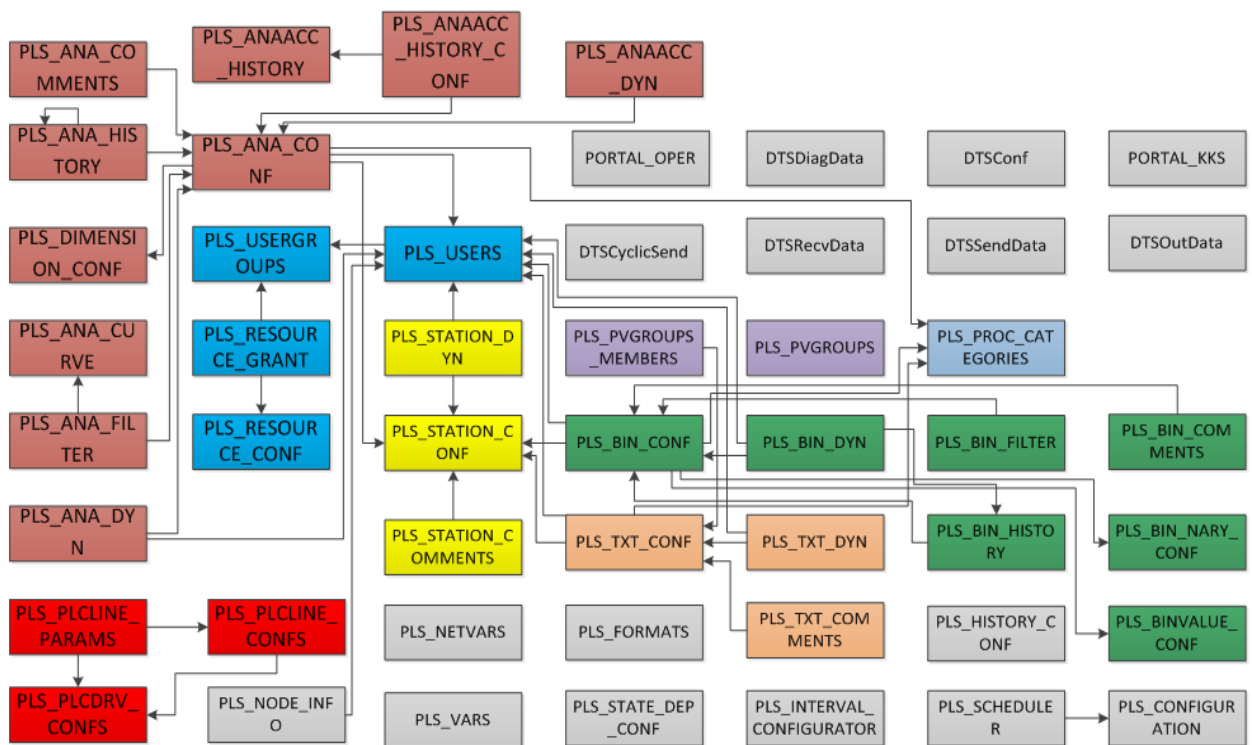


Рис. 6. Структура ОЗУ-резидентной базы данных реального времени

Рассматриваемая база данных характеризуется следующими свойствами:

- существует возможность обеспечить в реальном времени более 10 000 000 одиночных доступов в секунду;
- данные отображаются в таблицах со строками и столбцами;
- каждый из индивидуальных сетевых узлов в распределенной системе может хранить локальную копию таблиц базы данных. Репликация таблиц будет произведена автоматически в случае их изменения с помощью широкощитательных протоколов;
- пусковые механизмы позволяют какому-либо приложению определять действия, которые будут активированы автоматически, когда некое приложение меняет данные в базе данных;
- таблица базы данных может быть распределена на несколько серверов, каждый из которых отвечает за определенную часть данных.

Схема хранения распределенных объектов приведена на рис.7.

Таблицы базы данных располагаются в памяти компьютера для ускорения доступа к данным, однако имеется возможность получить файловое представление таблиц в виде дампов.

Дамп является файлом и представляет собой таблицу базы данных в текстовом формате. Одна строка файла соответствует одной строке таблицы. Первая строка файла содержит заголовок таблицы. Следующие строки определяют значения и конфигурации переменных, в соответствии с которыми они обрабатываются в разрабатываемой системе.

База данных формируется на основании импортных файлов низовой автоматик.

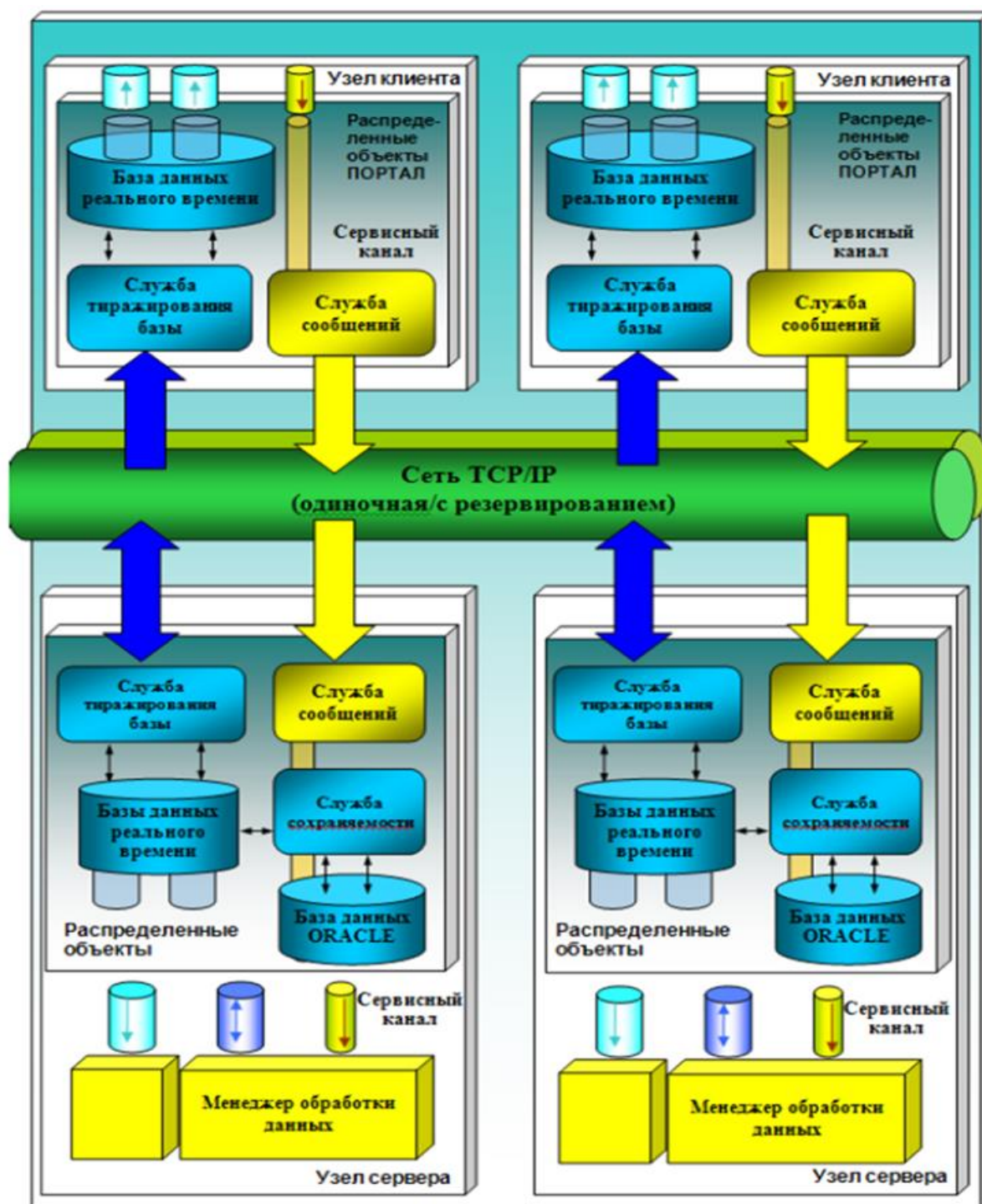


Рис.7. Структурная схема хранения и взаимодействия распределенных объектов

Языком программирования для написания кода программы был выбран Python. Алгоритм работы человеко-машинного комплекса представлен на рис. 8, принципиальная схема информационно-вычислительного комплекса – на рис. 9.

Таким образом, создан программный комплекс для проверки динамических связей технологических схем и баз данных, использующий систему кодирования Kraftwerk Kennzeichen System, позволяющий получать идентификаторы для типовых объектов энергоблока и выявляющий идентификаторы, которые не соответствуют базам данных.

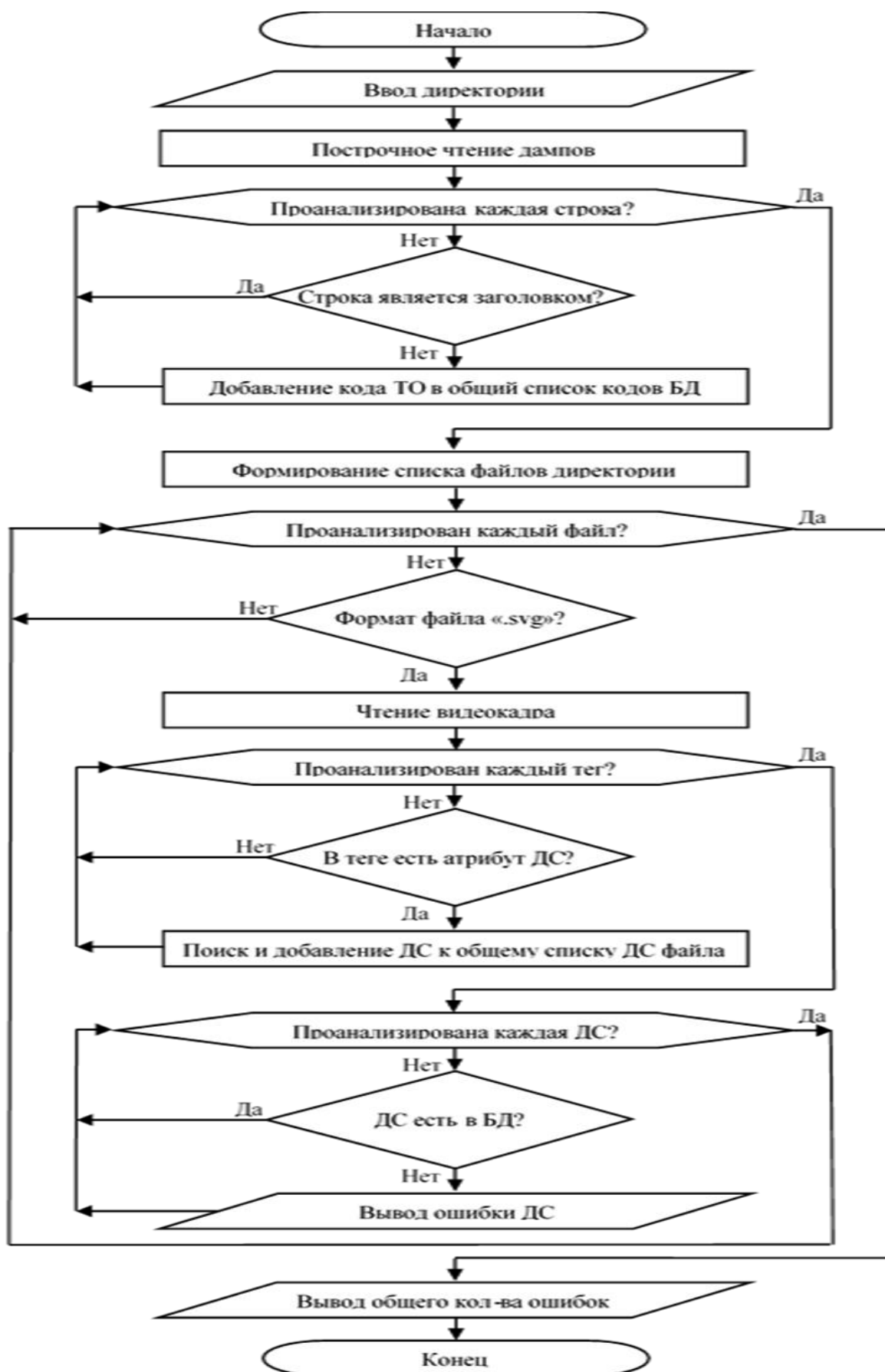


Рис.8. Структурная схема алгоритма работы человеко-машинного комплекса

При выборе оборудования для создания новой ИВС модернизируемого энергоблока был реализован принцип унификации. Программное обеспечение всех компонентов программно-технического комплекса (ПТК) ИВС, включая функции визуального контроля, реализовано на единых программных средствах. Представление значений параметров сигналов на всех рабочих станциях ПТК ИВС, интерфейсы взаимодействия, человеко-машинный интерфейс и навигация по видеокадрам идентичны, что учитывает требования по оптимальному взаимодействию системы «человек-машина» (рис. 9).

Система удовлетворяет требованиям по обеспечению надёжности на основе резервирования, независимости, разнообразия, с учётом отказов по общей причине. Для этого ИВС была реализована в виде двухканальной информационной системы. Основной и дублирующий каналы измерения и обработки данных в ПТК ИВС функционируют одновременно в полном объеме. В серверах оперативного контура вычислительного комплекса ИВС, к которым относятся два сервера ИВС и два сервера визуального контроля (СВРК), информация, полученная от двух каналов устройств связи с объектом: УСО ИВС и УСО СВРК обрабатывается совместно, что обеспечивает отсутствие влияния отказа в одном из каналов УСО на результаты работы ИВС.

Модульный подход позволяет иметь всего один дистрибутив программного обеспечения, а пакет настраиваемых «профилей» оборудования позволяет оперативно конфигурировать один «системный блок» под любые задачи, от шлюза до сервера. Эти свойства программного обеспечения дают возможность иметь универсальный резерв оборудования при ограниченном количестве ЗИП.

Таким образом, создано специальное программное обеспечение распределенной информационно-вычислительной системы управления, в которой шлюзовое прокладное программное обеспечение предусматривает до 40 протоколов обмена со смежными модулями, а также позволяет оптимизировать состав резервного оборудования и избежать программных конфликтов при обмене данными.

ЗАКЛЮЧЕНИЕ

В процессе выполнения диссертационного исследования получены следующие основные результаты:

1. Разработан механизм распределения данных о специальных транзакциях с оперативным контентом, обеспечивающий минимизацию времени передачи данных.
2. Разработан алгоритм локального кэширования хронологически запрошенных данных транзакций, обеспечивающий уменьшение задержку передачи данных о специальных транзакциях.
3. Разработан алгоритм разделения и перенаправления запросов между клиентами, межобъектными интерфейсами или облаком, дополнительно учитывающий интегрированную информацию о состоянии кэширования, предполагаемом размере данных и пропускной способности.
4. Модифицирован протокол планирования транзакций в СУБД реаль-

ного времени для учета не только временных ограничений транзакций, но и критериев, установленных пользователями базы данных.

5. Предложена архитектура СУБД реального времени с использованием модифицированного протокола планирования транзакций для улучшения качества предоставления услуг пользователям при реализации управления с обратной связью.

6. Создан программный комплекс для проверки динамических связей технологических схем и баз данных на основе системы кодирования Kraftwerk Kennzeichen System.

7. Созданы компоненты распределенной специальной информационно-вычислительной системы управления с унифицированными механизмами межмодульного взаимодействия для оптимизации состава резервного оборудования и исключения программных конфликтов при обмене данными.

8. Элементы программного обеспечения зарегистрированы в ФИПС.

Основные результаты диссертации опубликованы в следующих работах:

Публикации в изданиях, рекомендуемых ВАК РФ

1. Данилов, А.Д., Подход к управлению транзакциями в гетерогенных распределенных реплицированных системах баз данных в реальном масштабе времени / А.Д. Данилов, Д.С. Синюков Д.С // Системы управления и информационные технологии, 2021.- №3(85). – С. 59-65.

2. Синюков, Д.С. Разработка распределенной информационно-вычислительной системы управления ядерным блоком на Нововоронежской АЭС / Д.С. Синюков, А.Д. Данилов, А.А. Самодеенко, А.А. Иванников // Вестник Воронежского государственного технического университета, 2021.- т. 17, №6. – С. 20-27.

3. Синюков, Д.С. Модифицированный протокол планирования для оптимизации выполнения транзакций без превышения их предельных сроков / Д.С. Синюков // Системы управления и информационные технологии, 2021.- №3(89). – С. 89-94.

4. Синюков, Д.С. Экспериментальное исследование системы автоматического поиска и устранения неисправностей в базе данных / Д.С. Синюков, А.В. Потудинский // Моделирование, оптимизация и информационные технологии.- 2022. - 10(1). <https://moitvvt.ru/ru/journal/pdf?id=1150>. DOI: 10.26102/2310-6018/2022.36.1.030.

5. Синюков, Д.С. Программный комплекс для проверки динамических связей технологических схем и баз данных/ Д.С. Синюков, А.Д. Данилов, Д.А. Денисов, М.Е. Ушков// Вестник Воронежского государственного технического университета, 2022.- Т. 18. № 4.- С. 15-24.

Публикации в изданиях, входящих в базу цитирования Scopus

6. Sinyukov, D.S. Comparative analysis of software systems for intelligent operator support at the upper management level / D.S. Sinyukov, A.D. Danilov, I.A. Aksenov // AIP Conference Proceedings **2647**, 070009 (2022); <https://doi.org/10.1063/5.0104414> (Scopus).

Свидетельства о государственной регистрации программ для ЭВМ

7. Программа мониторинга серверов баз данных с оповещением средствами мессенджера / Н.И. Гребенникова, Д.С. Синюков, А.В. Потудинский: Свидетельство о регистрации программы для ЭВМ № 2022665036 от 09.08.2022. М.: ФИПС, 2022.

8. Программа анализа динамических связей интерактивных схем и баз данных реального времени / Д.С. Синюков, Д.А. Денисов, М.Е. Ушков: Свидетельство о регистрации программы для ЭВМ № 2022665490 от 17.08.2022. М.: ФИПС, 2022.

Статьи и материалы конференций

9. Данилов, А.Д. Механизм распределения данных о специальных транзакциях с оперативным контентом в реальном времени на основе кэширования в гетерогенных объектах распределенной сети / А.Д. Данилов, Д.С. Синюков // Информационные технологии моделирования и управления, 2021.- №3(125).- С. 216-223.

10. Синюков, Д.С. Применение систем управления базами данных как сервиса в сложных информационных системах / Д.С. Синюков, А.Д. Данилов // Тр. Всеросс. науч. Конф. «Достижения науки и технологий-ДНиТ-2021». – Красноярск, 2021. http://ru-conf.domnit.ru/media/filer_public/42/d3/42d35c66-d78a-411c-b74d-9ef2f99ff7b6/3006-dnit-2021.pdf.

11. Sinyukov, D.S. Problems of troubleshooting in databases / D.S. Sinyukov // Modern informatization problems in simulation and social technologies (MIP-2022'SCT): Proceedings of the XXVII-th International Open Science Conference (Yelm, WA, USA, January 2022). - Yelm, WA, USA: Science Book Publishing House, 2022. – p. 162-174.

12. Данилов, А.Д. Распределенная информационно-вычислительная система управления ядерным блоком / А.Д. Данилов, Д.С. Синюков // Завалишинские чтения'22: Сб. докл. XVII Междунар. конф. по электромеханике и робототехнике. - СПб., 2022. С. 33-37.

Подписано в печать 10.02.2023.

Формат 60x84/16. Бумага для множительных аппаратов.

Усл. печ. л. 1,0. Тираж 80 экз. Заказ №

ФГБОУ ВО «Воронежский государственный технический университет»
394006 Воронеж, 20-летия Октября, 84