

## МЕХАНИЗМЫ ПОДДЕРЖАНИЯ ВЫСОКОЙ ПРОИЗВОДИТЕЛЬНОСТИ ОБРАЗОВАТЕЛЬНЫХ ПЛАТФОРМ В УСЛОВИЯХ ПИКОВЫХ СЕЗОННЫХ НАГРУЗОК

Рамазанов Никита Максимович<sup>1</sup>

<sup>1</sup>Студент;  
НИТУ «МИСИС»;  
Россия, 119049, г. Москва, Ленинский пр-кт, д. 4, стр. 1;  
e-mail: nikramazan@mail.ru.

*В статье рассматривается проблема резкого роста нагрузки на образовательные платформы в пиковые периоды активности во время экзаменов и начала учебного года, что приводит к перебоям в работе сервисов. В ходе исследования проведен анализ основных причин деградации производительности и предложены механизмы поддержания стабильной работы системы: резервирование ресурсов, автоматическое масштабирование в облаке, трехуровневое кэширование и оптимизация стратегий хранения и обработки данных. Особое внимание уделено возможностям внедрения описанных механизмов в архитектуру существующих систем. Реализация предложенного решения позволит устранить проблемные места и увеличит надежность и производительность образовательной платформы, при этом экономично затрачивая ресурсы.*

**Ключевые слова:** образовательная платформа, микросервисная архитектура, производительность, масштабирование, резервирование ресурсов, облачные технологии, кэширование, хранение данных, обработка данных.

### Для цитирования:

Рамазанов Н. М. Механизмы поддержания высокой производительности образовательных платформ в условиях пиковых сезонных нагрузок// Системный анализ в науке и образовании: сетевое научное издание. 2025. № 1. С. 137-147. EDN: PVWMGB. URL: <https://sanse.ru/index.php/sanse/article/view/639>.

## MECHANISMS FOR ENSURING HIGH PERFORMANCE OF EDUCATIONAL PLATFORMS DURING PEAK SEASONAL LOADS

Ramazanov Nikita M.<sup>1</sup>

<sup>1</sup>Student;  
NUST MISIS;  
4/1 Leninskiy pr-kt, Moscow, 119049, Russia;  
e-mail: nikramazan@mail.ru.

*The study delves into the problem of a sharp load increase on educational platforms during peak periods of activity, such as exams and the beginning of the school year, which leads to service disruptions. The study analyzes the main causes of performance degradation and proposes mechanisms for maintaining stable system operation: resource reservation, automatic cloud-based scaling, three-level caching and optimization of data storage and processing strategies. Special attention is paid to the possibilities of implementing the described mechanisms into the architecture of existing systems. The implementation of the proposed solution will eliminate problem areas and increase the reliability and performance of the educational platform while using resources economically.*

**Keywords:** educational platform, microservice architecture, performance, scalability, resource reservation, cloud technologies, caching, data storage, data processing.

### For citation:

Ramazanov N. M. Mechanisms for ensuring high performance of educational platforms during peak seasonal loads, *System analysis in science and education*, 2025;(1):137-147 (in Russ). EDN: PVWMGB. Available from: <https://sanse.ru/index.php/sanse/article/view/639>.



Статья находится в открытом доступе и распространяется в соответствии с лицензией Creative Commons «Attribution» («Атрибуция») 4.0 Всемирная (CC BY 4.0) <https://creativecommons.org/licenses/by/4.0/deed.ru>

## Введение

Развитие информационных технологий и мировой курс на активную цифровизацию образования в настоящее время способствуют значительному росту сферы онлайн-обучения. Сильнейшим толчком данного явления стала пандемия *COVID-19*, когда учебные заведения повсеместно были вынуждены перевести занятия в дистанционный формат. Существующие на тот момент системы поддержки образовательного процесса были абсолютно не готовы к столь внезапному наплыву учащихся, а где-то их и вовсе не существовало [1].

За последние годы ситуация изменилась: появилось множество решений для организации дистанционного обучения, от простых платформ, точно закрывающих некоторые потребности, до комплексных систем со всеми необходимыми инструментами для наиболее эффективной подготовки, а также дополнительными возможностями, такими как применение искусственного интеллекта для персонализации образовательного процесса. Крупнейшие игроки рынка онлайн-образования демонстрируют впечатляющие показатели роста: их общая выручка увеличивается в среднем на 20-30% ежегодно [2].

Однако темпы роста числа пользователей и объема контента создают проблемы с отказоустойчивостью и масштабируемостью систем, что ставит под угрозу их стабильную работу. Особенно это заметно в моменты резких сезонных всплесков активности – например, в период экзаменов в школах и вузах, когда количество одновременных подключений может возрасти в 5-10 раз по сравнению с обычными днями.

Современные образовательные платформы сталкиваются с рядом технических вызовов:

- Необходимость обработки больших объемов учебного контента в реальном времени.
- Обеспечение стабильной работы систем при массовых входах.
- Управление потоками данных между серверами и клиентами.
- Поддержка интерактивных элементов обучения (онлайн-тестирований, виртуальных лабораторий).
- Хранение и обработка больших объемов аналитических данных об успеваемости.

Решение этих проблем требует комплексного системного подхода к архитектуре систем и внедрения современных технологий работы с высокими нагрузками. В данном исследовании будут рассмотрены ключевые механизмы обеспечения высокой производительности образовательных платформ в условиях пиковых сезонных нагрузок, проанализированы существующие решения и предложены подходы к организации масштабируемых систем дистанционного обучения, внедрение которых позволит улучшить пользовательский опыт за счет оптимизации ключевых метрик производительности и доступности.

Актуальность исследования обусловлена растущей зависимостью образовательной системы от цифровых технологий и необходимостью обеспечения бесперебойной работы платформ в критические периоды их использования.

## 1. Анализ проблематики объекта исследования

Объектом исследования являются информационно-образовательные системы (ИОС) – комплексные цифровые платформы, обеспечивающие организацию учебного процесса в режиме «онлайн», хранение и доставку образовательного контента, а также сбор и обработку данных об успеваемости пользователей.

Анализ практического опыта эксплуатации ИОС выявляет ряд критических проблем, требующих детального рассмотрения. В первую очередь, это недостаточная отказоустойчивость систем при пиковых нагрузках, что приводит к деградации качества обслуживания или полному отказу сервисов в ключевые моменты учебной деятельности [3].

В качестве корневой причины этого явления можно отметить сложности планирования подобных нагрузок – если общие тренды роста и падения трафика в течение года выявить еще можно, то такие события, как массовый перевод учащихся на «дистанционку» в период пандемии, предвидеть трудно [4]. Так, например, на рис. 1 видно, что платформа «Умскул», специализирующаяся на

подготовке учеников к школьным экзаменам, наименьшее число посетителей имеет в летние месяцы – что логично, поскольку это период каникул. А резкий рост аудитории наблюдается в сентябре – когда начинается учебный год – и в мае, перед самыми экзаменами.

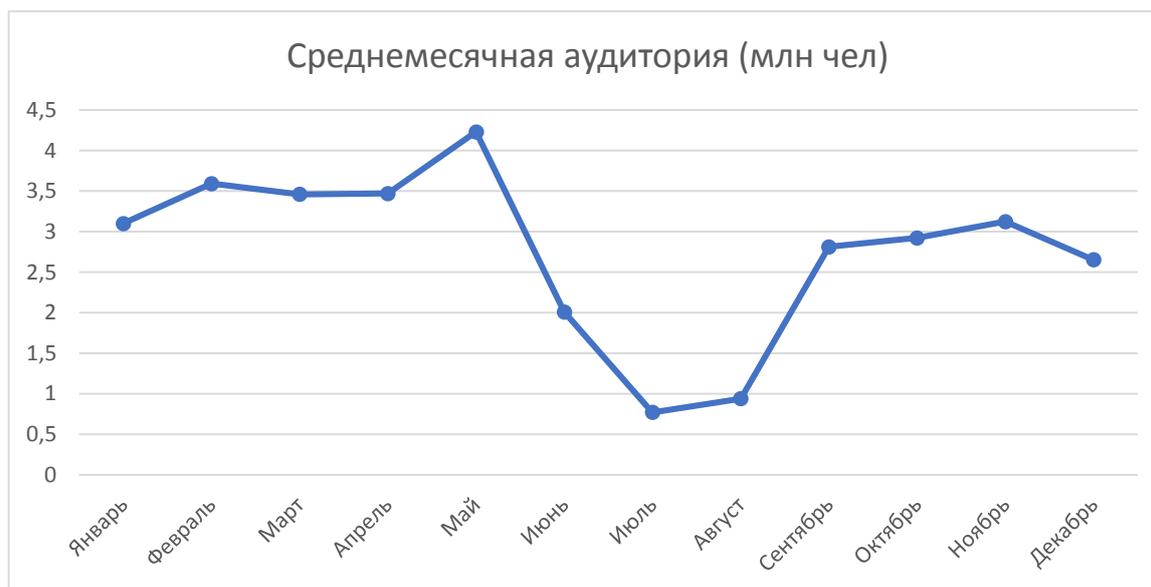


Рис. 1. Среднемесячная аудитория платформы «Умскул» за 2022-2024 гг.

В то же время, на примере образовательной платформы «Учи.ру» виден резкий скачок частоты поискового запроса «учи.ру» весной 2020-го года (рис. 2) – именно в тот момент школы массово перешли на дистанционный формат. Спрогнозировать подобный рост нагрузки на сервис не мог никто, как следствие – проблемы и перебои в обслуживании, вплоть до временной недоступности.



Рис. 2. Частота запроса «учи.ру» в поисковой системе Яндекс

Вдобавок к этому, существующие решения часто страдают от неэффективного использования вычислительных ресурсов. Традиционная практика выделения фиксированного объема серверов и хранилищ приводит либо к избыточным затратам при низкой загрузке, либо к критическим отказам в периоды высокой пользовательской активности. Это особенно важно в сфере онлайн-обучения, которая сильно подвержена влиянию сезонности.

Еще одной серьезной проблемой также является отсутствие развитой системы быстрого доступа к данным. Существующие решения либо ограничиваются базовым *HTTP*-кэшированием, либо используют разрозненные кэши на отдельных компонентах системы, что не обеспечивает оптимального баланса между производительностью и актуальностью данных. В области хранения данных наблюдается еще ряд сложностей, связанных с ростом объемов информации. Традиционные монолитные базы данных демонстрируют снижение производительности при достижении определенного порога данных, что приводит к увеличению времени отклика и усложнению процессов администрирования.

Отдельно стоит выделить отсутствие в некоторых решениях автоматизированных механизмов адаптации к изменяющимся нагрузкам. Администраторам приходится вручную масштабировать инфраструктуру, что приводит к задержкам в реагировании на рост нагрузки и дополнительным затратам на поддержку.

Все вышеперечисленные проблемы требуют комплексного подхода к их решению, учитывающего как текущие потребности образовательных организаций, так и перспективы дальнейшего развития цифровых образовательных технологий.

## 2. Предлагаемые решения

### 2.1. Резервирование ресурсов

Резервирование ресурсов является фундаментальной основой обеспечения бесперебойной работы любой высоконагруженной системы, в частности образовательной платформы. Данный подход предполагает создание избыточной инфраструктуры, способной взять на себя нагрузку при выходе из строя компонентов системы или при резком увеличении трафика.

В основе решения лежит принцип предварительного планирования мощностей с учетом прогнозируемых пиковых нагрузок. Это включает анализ исторических данных о посещаемости платформы, сезонных колебаний активности пользователей и тщательное планирование масштабных образовательных мероприятий.

Система резервирования должна охватывать все критически важные компоненты инфраструктуры, особенно серверы приложений и баз данных. Особое внимание уделяется созданию резервных копий данных, которые должны храниться на отдельных физических носителях и регулярно обновляться. Рекомендуется использовать иерархический подход к резервному копированию, включающий еженедельные полные бэкапы данных, ежедневные инкрементальные копии и периодическое создание архивных копий.

Такой комплексный подход к резервированию ресурсов обеспечивает:

- минимальное время восстановления после сбоев;
- защиту от потери данных: в случае сбоя будет потеряно минимальное количество данных;
- способность выдерживать кратковременные пики нагрузки;
- надежность работы платформы в долгосрочной перспективе.

При внедрении системы резервирования необходимо разработать детальный план аварийного восстановления, включающий процедуры переключения на резервные компоненты, восстановления из резервных копий и возвращения к штатной работе после устранения неисправностей. При этом она должна быть интегрирована с общей системой мониторинга платформы, что позволит оперативно выявлять потенциальные проблемы и предотвращать их развитие до критического состояния.

Эффективность предложенного решения может быть оценена по показателям доступности системы (*uptime*), времени восстановления после сбоев (*RTO*) и допустимой потери данных (*RPO*). Рекомендуется устанавливать целевые значения этих показателей исходя из требований к надежности конкретной образовательной платформы.

### 2.2. Автоматическое облачное масштабирование

Автоматическое облачное масштабирование представляет собой современный подход к управлению ресурсами информационной системы, позволяющий динамически адаптироваться к изменяющимся нагрузкам. Данный метод базируется на принципах *elastic computing*, где вычислительные ресурсы могут гибко расширяться или сокращаться в зависимости от текущих потребностей системы. Ведущие облачные провайдеры, такие как «Yandex Cloud», предоставляют обширные возможности для реализации описанных принципов в рамках любой системы, в том числе ИОС.

В основе решения лежит использование «автоскейлеров» – специальных компонентов, отслеживающих метрики производительности и принимающих решения о масштабировании. Процесс

принятия этого решения представлен на рис. 3. Автоскейлер анализирует загрузку *CPU*, объем используемой памяти, количество одновременных соединений и время обработки запросов. На основе этих данных система автоматически запускает или останавливает экземпляры (инстансы) приложений.

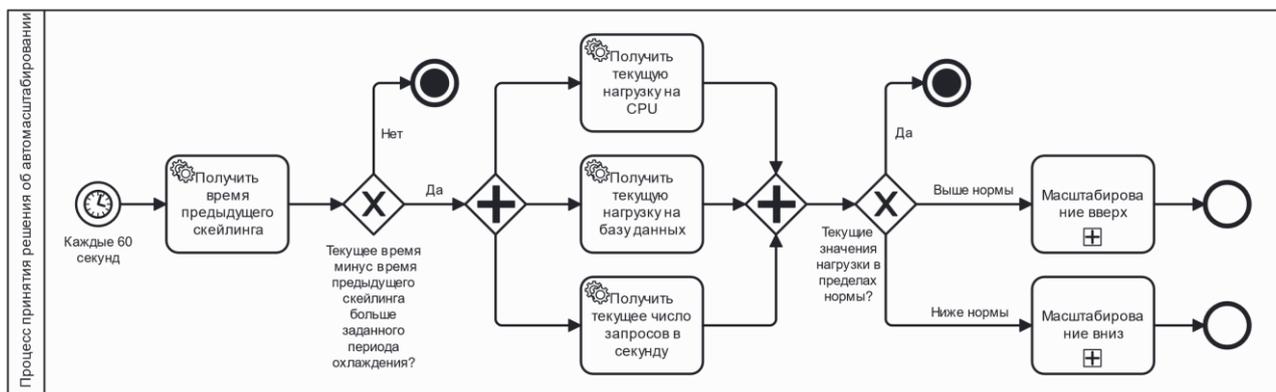


Рис. 3. Процесс принятия решения о масштабировании

Правила, по которым принимаются данные решения, называют политиками масштабирования. Они могут определять:

- минимальное и максимальное количество экземпляров приложения;
- условия для масштабирования вверх или вниз;
- период охлаждения между масштабированиями (задержка, позволяющая системе стабилизироваться во время пиков нагрузки);
- приоритеты использования различных типов инстансов.

Ключевым элементом является внедрение контейнерной оркестрации в облаке с использованием технологии *Kubernetes*. Это позволяет эффективно управлять жизненным циклом контейнеров, обеспечивая автоматическую перезагрузку при сбоях и оптимизацию использования ресурсов. *Kubernetes*-кластеры автоматически масштабируются в зависимости от текущей нагрузки, что гарантирует стабильную работу платформы при любом количестве пользователей.

Основное преимущество такого подхода заключается в существенной экономии средств – ресурсы подключаются только по необходимости, и как только такая необходимость пропадает – отключаются. Помимо финансовой выгоды, данный механизм увеличивает надежность системы, снижает риски простоев и их время, а также уменьшает влияние человеческого фактора, поскольку наиболее критичные процессы автоматизированы [5, 6]. Автоматическое облачное масштабирование позволяет создать гибкую и экономически эффективную инфраструктуру, способную адаптироваться к любым изменениям нагрузки без необходимости ручного вмешательства администраторов.

Важно отметить необходимость тщательной настройки политик масштабирования с учетом специфики образовательной платформы и характера нагрузки. Рекомендуется провести серию нагрузочных тестов для определения оптимальных параметров автоскейлинга, в противном случае система может столкнуться как с неэффективным использованием выделенных ресурсов, так и с их недостатком в пиковые периоды.

### 2.3. Трехуровневое кэширование

Для обеспечения высокой производительности образовательной платформы предлагается реализовать многоуровневую систему кэширования, которая позволит существенно снизить нагрузку на основные компоненты инфраструктуры и ускорить обработку запросов пользователей. Для образовательных платформ особенно важно кэшировать учебные материалы (учебники, статьи, условия задач), медиаконтент (изображения, видеоролики, аудиоролики), стили и скрипты для быстрой загрузки компонентов веб-страниц, а также результаты вычислений, например, количество заданий в онлайн-тестировании.

Первый уровень – это уровень *CDN* (*Content Delivery Network*). Он реализуется через использование глобальной сети доставки контента. *CDN*-сеть размещает статический контент на геоуровнеженных узлах, расположенных ближе к конечным пользователям. Схема взаимодействия пользователей с *CDN* представлена на рис. 4. Это позволяет сократить время загрузки контента, обеспечить его высокую доступность и снизить нагрузку на основные вычислительные мощности серверов. Для целей образовательной платформы можно воспользоваться услугами существующих провайдеров *CDN*-серверов, имеющих широкое покрытие по всей территории РФ и не только.

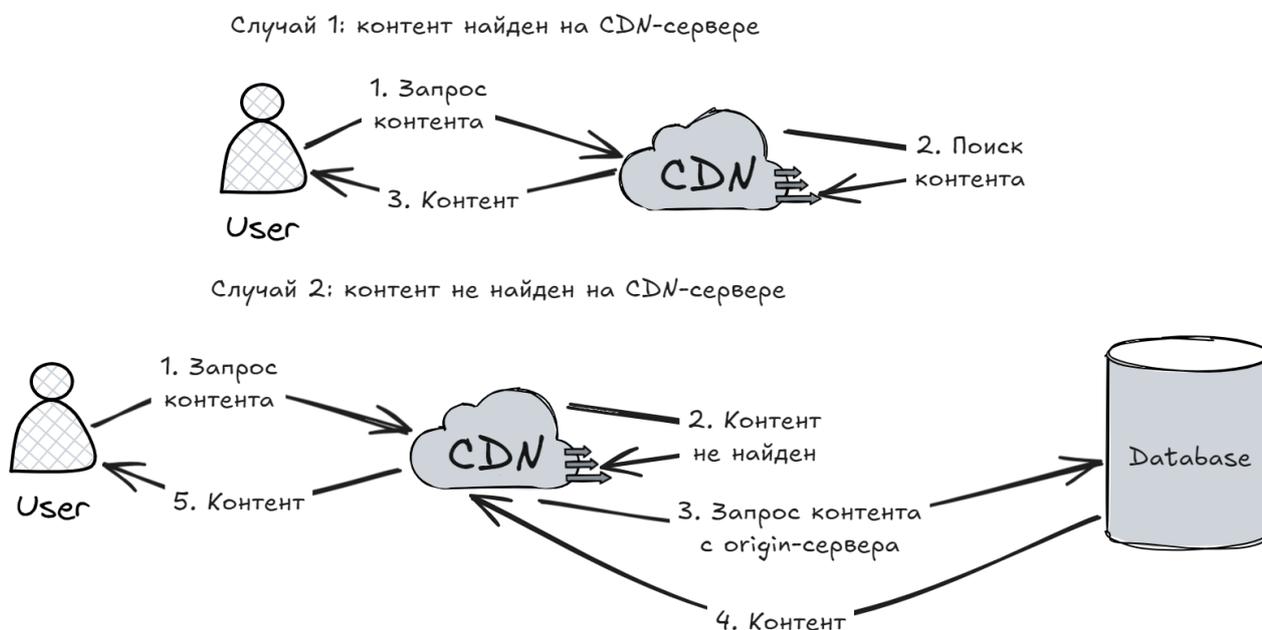


Рис. 4. Схема взаимодействия пользователей с *CDN*

Второй – уровень клиентского кэширования. Он реализуется на стороне клиента и включает:

- кэширование статических ресурсов (визуального оформления страницы, анимации) в браузере;
- локальное хранение медиафайлов: предварительная загрузка изображений, аудио и видео;
- офлайн-режим: возможность доступа к некоторому контенту без подключения к сети;
- сжатие и минимизацию передаваемых данных для экономии сетевого трафика и ускорения загрузки.

Также стоит рассмотреть возможность кэширования клиентских сессий – это позволит разгрузить сервисы аутентификации, благодаря чему платформа сможет выдерживать большее количество одновременных подключений.

И, наконец, третий уровень – уровень кэширования запросов. Третий уровень реализуется на стороне сервера приложения и является последним рубежом перед запросом непосредственно в базу данных. Обычно его используют для сохранения результатов сложных запросов, которые долго выполняются и нечасто меняются, но при этом часто используются. Например, вместо того чтобы каждый раз при входе на платформу запрашивать в базе данных успеваемость ученика для отображения в интерфейсе – можно сохранить ее в кэше для быстрого доступа и обновлять только тогда, когда успеваемость изменяется после выполнения тестов или просмотра занятий. Варианты стратегий кэширования представлены на рис. 5.

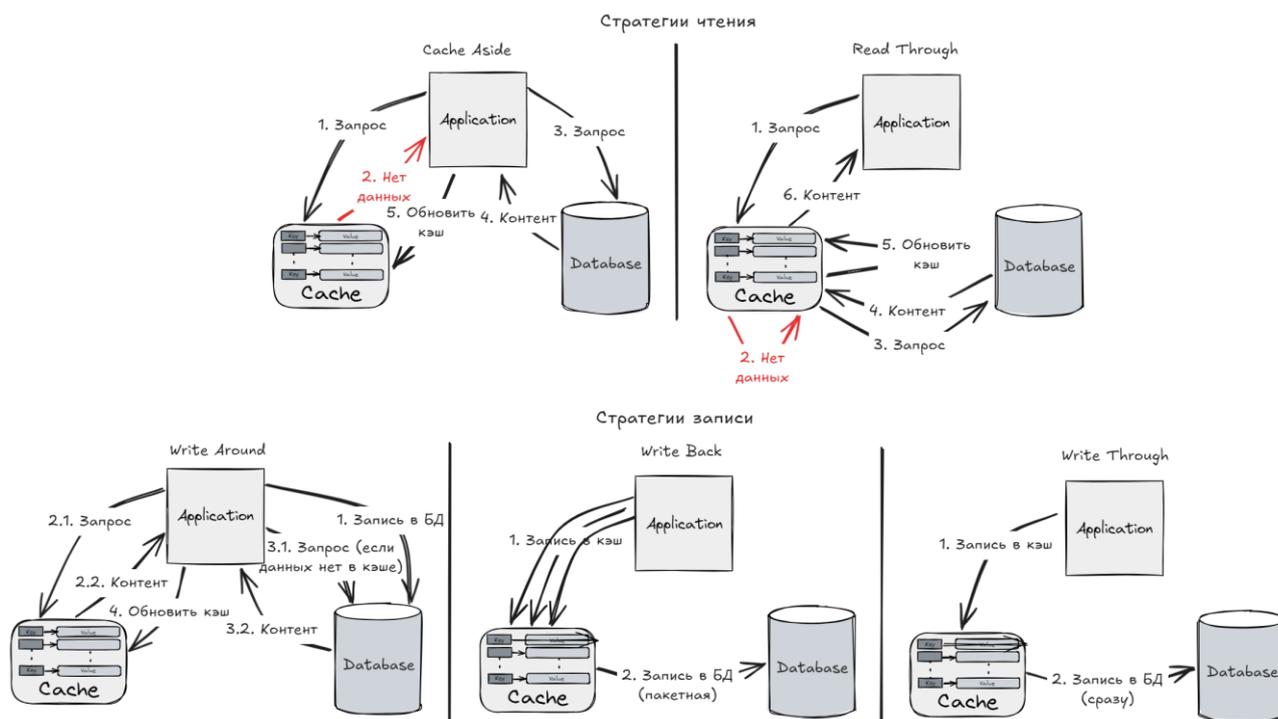


Рис. 5. Варианты стратегий кэширования

Однако мало реализовать все три уровня – успешная реализация трехуровневой системы кэширования требует согласованной работы всех уровней, при этом должны быть выбраны подходящие для каждого конкретного случая стратегии кэширования и политики его обновления.

Эффективность предложенной системы кэширования может быть оценена главным образом по сокращению времени отклика системы на запросы, которое можно измерить, проведя ряд тестов. В качестве сопутствующих преимуществ стоит выделить снижение нагрузки на серверные компоненты и увеличение числа одновременно обрабатываемых запросов.

## 2.4. Оптимизация стратегий хранения и обработки данных

Эффективное управление данными является критически важным аспектом обеспечения производительности образовательной платформы. Для достижения оптимальной производительности предлагается комплексный подход к организации хранения данных, включающий несколько взаимосвязанных механизмов.

В основе решения лежит принцип партиционирования данных, который позволяет разделить большие таблицы на более мелкие, управляемые части. Это существенно улучшает производительность при выполнении операций чтения и записи, а также упрощает резервное копирование и восстановление данных. Партиционирование осуществляется по ключевым атрибутам, таким как дата создания записи, идентификатор курса или пользователя, что обеспечивает равномерное распределение нагрузки.

Дополнительным уровнем оптимизации служит технология шардирования, которая предполагает разделение данных на независимые части (шарды) с последующим размещением на разных физических узлах. Это позволяет горизонтально масштабировать систему хранения и обрабатывать большие объемы данных с высокой производительностью. При проектировании схемы шардирования учитывается характер доступа к данным и требования к производительности различных типов запросов.

Репликация данных – еще один, весьма популярный механизм для обеспечения высокой доступности данных. Самая частая реализация этого механизма – это режим «*master-slave*», или «лидер-подчиненный»: есть главный экземпляр базы данных, в котором выполняются операции записи данных. После записи данные автоматически копируются во все подчиненные БД (реплики) согласно определенной стратегии копирования. При этом на репликах возможно только чтение данных, а на

«лидере» - как чтения, так и записи. В случае сбоя система автоматически назначает нового лидера, гарантируя непрерывность работы. В нормальном режиме работы такой подход позволяет разгрузить основную базу от операций чтения, что позволит оптимизировать ее специально под операции записи и таким образом ускорить их обработку.

Еще одним важным аспектом является организация хранения различных типов данных с учетом специфики их использования. Для проведения транзакций оплаты курсов подойдет только реляционная база данных (например, *PostgreSQL*) со строгим соблюдением согласованности; в случае каталога курсов – прекрасным решением будет нереляционная документоориентированная база данных, такая как *MongoDB*; для двоичных файлов с медиаконтентом лучше всего подойдет специализированное *S3*-хранилище с поддержкой потоковой передачи данных. Наконец, стоит разделять базы данных, используемые непосредственно для работы системы, и хранилища данных для аналитических нужд – для этого используют оптимизированные инструменты специально под такие задачи, например, это могут быть *Clickhouse* и *Greenplum*.

Для управления жизненным циклом данных должно применяться иерархическое хранение с автоматическим перемещением устаревших данных на менее производительные, но более экономичные носители. Это позволяет оптимизировать затраты на хранение при сохранении высокой производительности доступа к актуальным данным.

Все компоненты системы хранения интегрируются в единую архитектуру (рис. 6), обеспечивающую согласованную работу и эффективное использование ресурсов. Это достигается за счет централизованного управления конфигурацией, автоматического распределения нагрузки и координации между различными уровнями хранения данных.

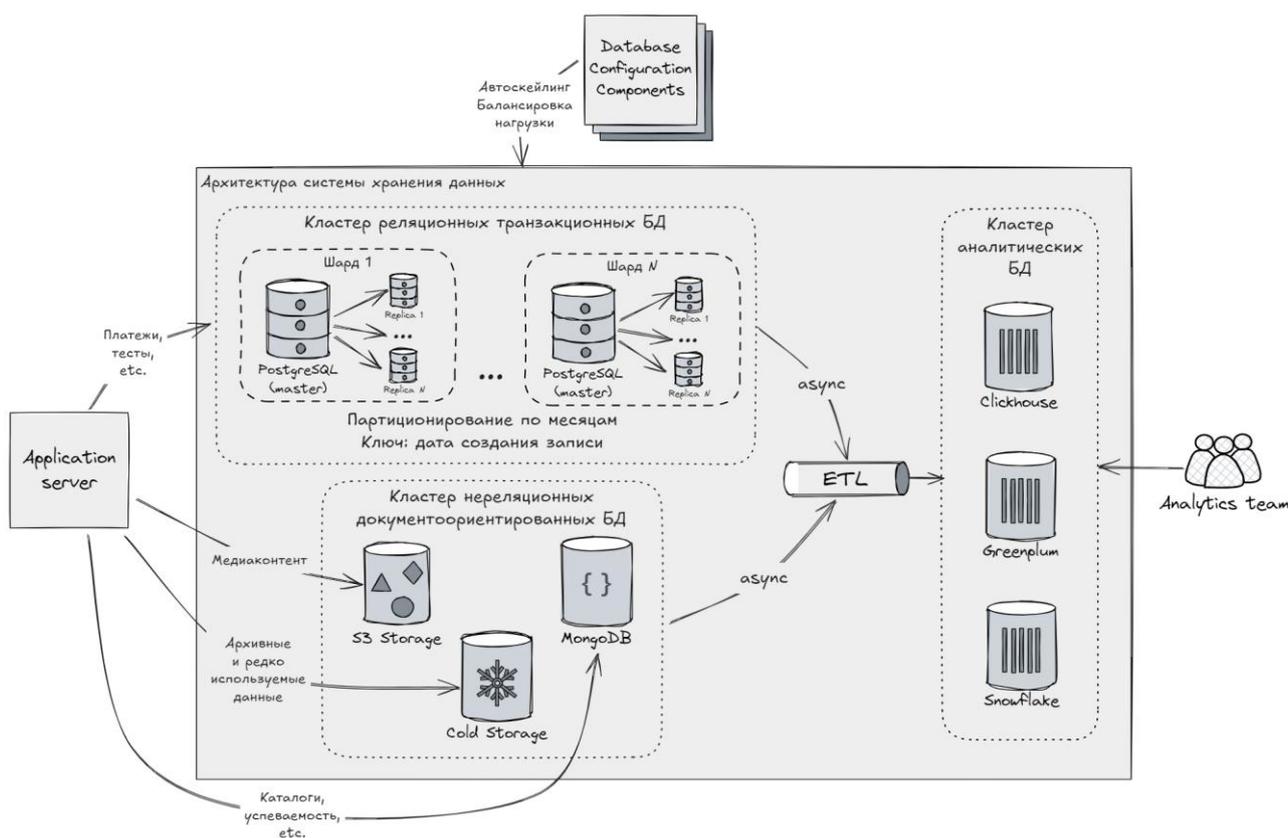


Рис. 6. Архитектура системы хранения и обработки данных

Внедрение описанного подхода к организации хранения данных позволяет создать масштабируемую и высокопроизводительную инфраструктуру, способную эффективно справляться с растущими объемами данных и требованиями к производительности образовательной платформы.

### 3. Целевая архитектура

Все предложенные решения взаимосвязаны и должны внедряться комплексно для достижения максимальной эффективности. Их реализация позволит создать высокопроизводительную образовательную платформу, способную эффективно работать и выдерживать возможные нагрузки, в том числе во время резких сезонных или внеплановых скачков трафика. Целевая архитектура такой платформы представлена на рис. 7. Для ее описания выбрана специализированная нотация C4.

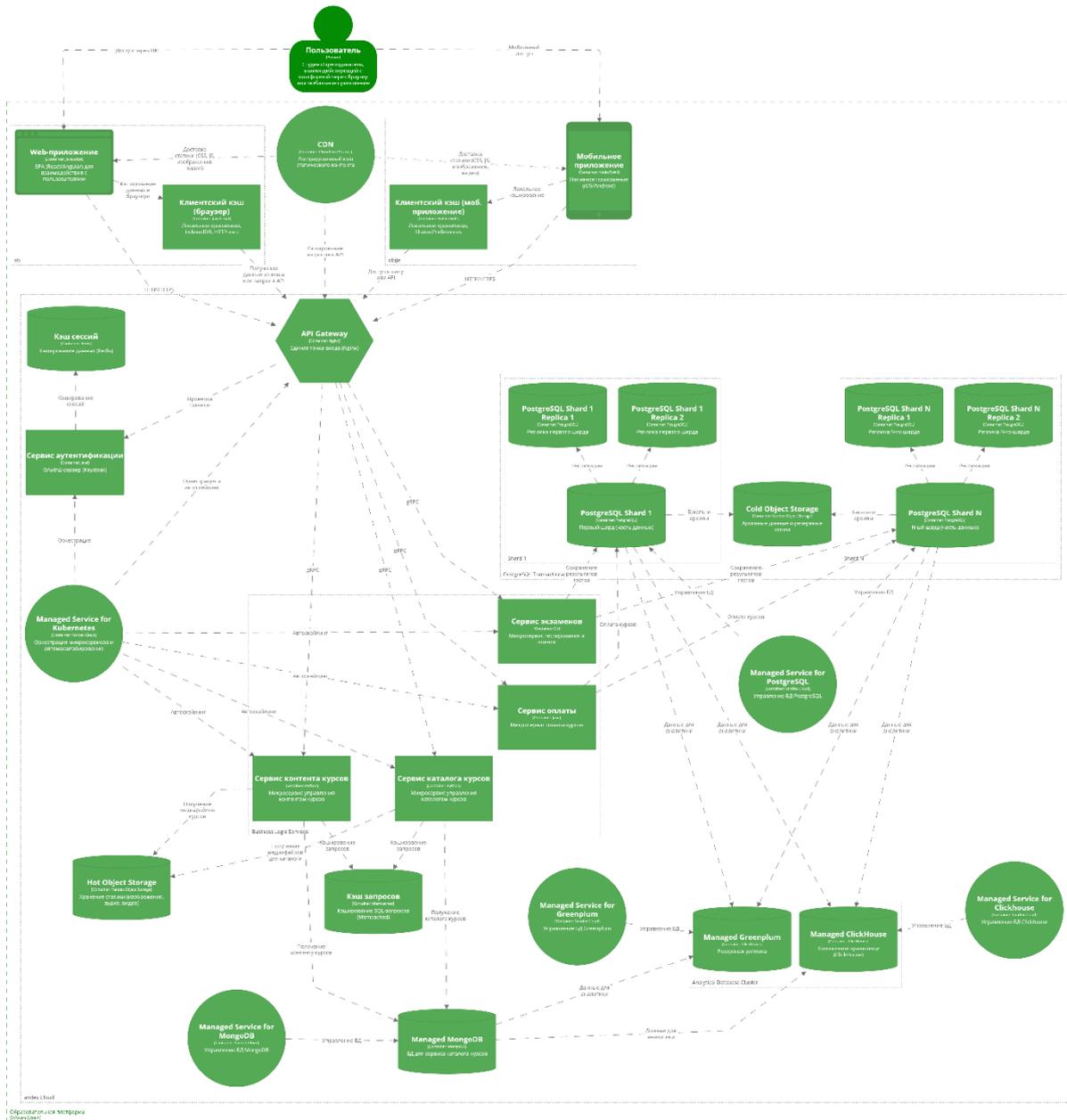


Рис. 7. Целевая архитектура образовательной платформы

При внедрении данных решений также необходимо учитывать специфику конкретной системы и проводить тщательное тестирование для оптимизации параметров работы каждого компонента. Кроме этого, стоит отметить, что на ранних этапах развития ИОС не требуется сразу внедрять все описанные механизмы – это будет неоправданно и может не окупиться. Полномасштабное внедрение рекомендуется только для зрелых систем, которые уже имеют стабильный клиентский трафик и готовы к потенциальным временным проблемам в процессе внедрения ради достижения в конечном счете

преимуществ предложенной архитектуры. Развивающимся же системам стоит двигаться в этом направлении постепенно.

## Заключение

В результате проведенного исследования были выявлены ключевые проблемы, связанные с обеспечением высокой производительности образовательных платформ в периоды пиковых сезонных нагрузок. Установлено, что существующие решения часто не справляются с растущими требованиями к масштабируемости и отказоустойчивости систем.

Проведенное исследование позволило разработать комплексный подход к решению выявленных проблем, включающий следующие ключевые механизмы:

- Систему резервирования ресурсов для гарантированного обеспечения производительности и работоспособности системы.
- Автоматическое облачное масштабирование для динамической адаптации к нагрузкам и экономии вычислительных и финансовых ресурсов.
- Трехуровневое кэширование для ускорения доступа к контенту.
- Оптимизацию стратегий хранения и обработки данных для быстрой и надежной обработки запросов пользователей.

Практическая значимость исследования заключается в создании конкретных рекомендаций по модернизации существующих образовательных платформ и по реализации новых. Предложенные решения могут быть адаптированы под различные типы ИОС с учетом их специфики и масштаба.

Перспективы дальнейших исследований могут быть связаны с:

- Разработкой системы прогнозирования пиковых нагрузок на основе машинного обучения.
- Исследованием новых методов оптимизации хранения данных.
- Созданием интеллектуальных систем управления ресурсами.
- Разработкой методик оценки эффективности внедренных решений.

Реализация предложенных механизмов позволит создать новое поколение образовательных платформ, способных эффективно функционировать в условиях растущих требований к их производительности и надежности.

## Список источников

1. Назаров В. Л. Анализ первых результатов цифровой трансформации школьного образования в период мировой пандемии covid-19: опыт Свердловской области : монография / В. Л. Назаров, Л. И. Долинер ; Министерство науки и высшего образования Российской Федерации, Уральский федеральный университет имени первого Президента России Б. Н. Ельцина. — Екатеринбург : Издательство Уральского университета, 2021. — 148 с.
2. Рейтинг крупнейших компаний на рынке онлайн-образования / Smart Ranking. – [Обновляется в течение суток.] – URL: <https://edtechs.ru/> (дата обращения: 16.02.2025).
3. Первый замглавы Пскова: Сбои в работе «Сферума» обусловлены пиковой нагрузкой // Псков. Великие Луки. Новости Пскова и Псковской области : [портал]. – 2024. – URL: <https://pln-pskov.ru/society/534464.html> (дата обращения: 18.02.2025).
4. Паршин И. Илья Паршин — РБК: «Пандемия даст мощный толчок развитию EdTech-отрасли» // РБК Тренды. – ООО «БИЗНЕСПРЕСС», АО «РОСБИЗНЕСКОНСАЛТИНГ», 1995–2025. – Дата обновления: 27.11.2020. – URL: <https://trends.rbc.ru/trends/education/5ea6eb949a7947a20c816f41> (дата обращения: 18.02.2025).
5. Как платформу «Открытое образование.рф» готовили к сверхвысоким нагрузкам – Истории успеха // Yandex Cloud. – ООО «Яндекс.Облако», 2025. –Дата публикации: 20.07. 2021. – URL: <https://yandex.cloud/ru/cases/openedu> (дата обращения: 19.02.2025).

6. Как обеспечить бесперебойную работу обучающей платформы при скачках трафика: кейс «Учи.ру» // Инновационный центр «Сколково». – Официальный сайт Фонда «Сколково», 2025.- Дата публикации: 4.04.2022. – URL: <https://sk.ru/news/kak-obespechit-besperebojnuyu-rabotu-obuchayushej-platformy-pri-skachkah-trafika-kejs-uchiru/> (дата обращения: 19.02.2025).
7. Как подготовить ИТ-инфраструктуру к пиковым сезонным нагрузкам / ITGLOBAL.COM // Компьютерра. – Компьютерра, 1997–2025. – Дата публикации: 09.11.2022. – URL: <https://www.computerra.ru/283466> (дата обращения: 19.02.2025).