



ПОИСК ПРОТОТИПА И КАСКАДНОЕ КОПИРОВАНИЕ В РАДИОЧАСТОТНОМ РЕЕСТРЕ: ПРОИЗВОДИТЕЛЬНОСТЬ И ОПТИМИЗАЦИЯ ЗАПРОСОВ

¹Аданбаев А. М. ORCID ID 0009-0005-5888-1220,

²Сагымбаев А. А. ORCID ID 0009-0009-6177-0627

¹ *Киргизско-Турецкий университет «Манас», Бишкек,
Киргизская Республика, e-mail: aybek@manas.edu.kg;*

² *Киргизский государственный технический университет
им. И. Раззакова, Бишкек, Киргизская Республика*

Ведение радиочастотного реестра включает повторяющиеся операции, при которых оператору приходится заново вводить уже известные сведения. Цель статьи – формализовать механизм поиска прототипа и каскадного копирования типовых записей, а также экспериментально оценить производительность поиска и эффект оптимизации запроса. Исследование выполнено на агрегированных данных действующего радиочастотного реестра Киргизской Республики объёмом 667 193 записи в среде реляционной системы управления базами данных; использовались замеры времени выполнения запросов, анализ фактического плана выполнения и статистическая оценка доверительного интервала. Показано, что последовательное уточнение критериев сокращает множество записей-кандидатов от полного объёма реестра до нуля при времени отклика в пределах нескольких десятков миллисекунд. Анализ плана выполнения выявил причину аномальной задержки в режиме обзорного просмотра: отсутствие индекса по полю регионального кода приводило к полному сканированию промежуточной таблицы и многократному запуску зависимого подзапроса. Добавление индекса устранило полное сканирование и обеспечило ускорение в 6,2 раза, подтверждённое измерениями времени выполнения и анализом плана запроса. Обобщение результата ограничено системами со сходной структурой запроса: категориальным фильтром верхнего уровня, зависимым подзапросом и отсутствующим индексом по полю фильтрации. Подсчёт атрибутов схемы показывает, что каскадное копирование сокращает число потенциально заполняемых оператором полей примерно с 50 до 15; фактическое время работы оператора и частота ошибок ввода в данной статье не измерялись.

Ключевые слова: радиочастотный реестр, управление спектром, поиск прототипа, копирование записей, оптимизация запросов, индексирование базы данных, зависимый подзапрос

PROTOTYPE SEARCH AND CASCADING COPY IN A RADIO FREQUENCY REGISTER: PERFORMANCE AND QUERY OPTIMISATION

¹Adanbaev A. M. ORCID ID 0009-0005-5888-1220,

²Sagymbaev A. A. ORCID ID 0009-0009-6177-0627

¹ *Kyrgyz-Turkish Manas University, Bishkek, Kyrgyz Republic,
e-mail: aybek@manas.edu.kg;*

² *Kyrgyz State Technical University named after I. Razzakov,
Bishkek, Kyrgyz Republic*

Maintenance of a radio frequency register includes recurring operations in which operators have to re-enter information that is already known. The aim of the paper is to formalise a prototype search and cascading copy mechanism for typical records and to experimentally evaluate search performance and the effect of query optimisation. The study was conducted on aggregated data from the active radio frequency register of the Kyrgyz Republic containing 667,193 records in a relational database management system environment; query execution time measurements, actual execution-plan analysis and statistical confidence interval estimation were used. The results show that successive refinement of search criteria reduces the candidate set from the full register volume to zero while keeping response times within a few tens of milliseconds. Execution-plan analysis identified the cause of the anomalous delay in the overview mode: the absence of an index on the regional code field led to a full scan of the intermediate table and repeated execution of a dependent subquery. Adding the index removed the full scan and provided a 6.2-fold speedup, confirmed by execution time measurements and query-plan analysis. Generalisation of the result is limited to systems with a similar query structure: an upper-level categorical filter, a dependent subquery and the absence of an index on the filtering field. A schema-level attribute count shows that cascading copy reduces the number of fields potentially filled in by an operator from about 50 to about 15; the actual operator working time and input error rate were not measured in this paper.

Keywords: radio frequency register, spectrum management, prototype search, record copying, database query optimisation, database indexing, dependent subquery

Введение

Автоматизированные системы управления радиочастотным спектром (АИС РЧС) являются базовым инструментом национальных регуляторов в области электросвязи. Рекомендация ITU-R SM.1370-2 задаёт основные требования к таким системам: состав данных, поддерживаемые функции и возможность выборки по диапазону частот, географической области, идентификатору записи и статусу лицензии [1; 2]. Документы МСЭ также рекомендуют использовать компьютеризированные методы управления спектром и соответствующие модели данных [3; 4]. Иными словами, современная АИС РЧС должна не только хранить сведения о лицензиатах, объектах размещения радиоэлектронных средств и параметрах частотных присвоений, но и обеспечивать удобную работу с этими данными [4].

Практика эксплуатации государственного реестра РЧС Киргизской Республики показывает, что значительная часть новых частотных присвоений оформляется для объектов, технические параметры которых либо совпадают с уже зарегистрированными, либо отличаются несущественно. В литературе по цифровизации государственных услуг повторный ввод данных и недостаточная стандартизация процедур рассматриваются как факторы, способные увеличивать административную нагрузку и снижать удобство работы с цифровыми сервисами [5–7]. Для систем государственного управления, ориентированных на интеграцию электронных услуг, отдельное значение имеют согласованная работа информационных контуров и межведомственное взаимодействие [8]. Принципы повторного использования уже имеющихся сведений и однократного предоставления данных рассматриваются как важные направления ограничения избыточного ввода и дублирования сведений [9; 10]. В настоящей статье указанные источники используются как общий контекст постановки задачи, а не как прямое доказательство эффективности каскадного копирования в радиочастотном реестре.

Несмотря на наличие международных руководств, в доступных рецензируемых публикациях по автоматизированному контролю назначения радиочастот и управлению ресурсами радиосистем ограниченно представлены прикладные сценарии ведения национального радиочастотного реестра, в которых поиск прототипа, создание записи на основе копии и анализ производительности рассматриваются совместно [11; 12]. В настоящей статье ко-

личественно оцениваются производительность поиска и эффект индексирования, а каскадное копирование рассматривается на уровне структуры создаваемых записей и числа потенциально заполняемых полей. Механизмы экспорта и печати описываются как элементы рабочего контура, но их производительность отдельно не измеряется. В этой связи статья вносит следующий вклад: 1) экспериментально измеряет производительность многокритериального поиска; 2) с помощью EXPLAIN ANALYZE диагностирует причину деградации режима byOne; 3) показывает эффект индексирования – ускорение в 6,2 раза при 95%-ном доверительном интервале 5,69–7,16 раза; 4) формулирует условия, при которых аналогичный эффект может ожидать в системах сходной архитектуры, без утверждения его универсальной воспроизводимости.

Цель исследования – формализовать механизм поиска прототипа и каскадного копирования при автоматизированном ведении радиочастотного реестра, экспериментально оценить производительность поиска и эффект индексирования, а также оценить каскадное копирование на уровне числа потенциально заполняемых оператором полей. Количественная оценка времени работы операторов, частоты ошибок ввода, а также производительности экспорта и подготовки печатных форм в задачи настоящего исследования не входила. Ниже приведены материалы исследования и использованные методы.

Материалы и методы исследования

Материал исследования включает три компонента: а) агрегированные данные действующего радиочастотного реестра объёмом 667 193 записи (MySQL 8.0.45; без учёта записей, помеченных через поле `addflag`); б) архивные данные операционного журнала предшествующей системы за период с июня 2000 г. по октябрь 2014 г.; в) программную реализацию механизмов АИС управления РЧС КР. Свидетельство о регистрации программы [13] используется только для идентификации программного решения как объекта исследования и не рассматривается как доказательство научной эффективности предложенных механизмов.

С точки зрения предметной области трёхуровневая организация сведений о лицензиате, объекте размещения радиоэлектронных средств и частотном присвоении согласуется с общей логикой данных автоматизированных систем управления радиочастотным спектром [1–3]. В исследуемой

реализации реестр построен на реляционной базе данных и включает три связанные таблицы. Таблица Owner описывает лицензиата и содержит около 20 атрибутов; поле area включает 9 целочисленных кодов (7 областей КР, а также Бишкек и Ош как города республиканского значения), а поле state хранит коды статуса лицензии. Таблица Site описывает объект размещения РЭС и содержит около 20 технических атрибутов; в ней 271 453 строки. Таблица Freq фиксирует частотное присвоение, содержит около 10 атрибутов, а номинал частоты хранится в кГц. Полное пространство поиска задаётся как

$$F = \text{Owner} \bowtie \text{Site} \bowtie \text{Freq}, |F| = 667\,193,$$

где \bowtie обозначает соединение таблиц, то есть формирование набора строк с совпадающими значениями в связанных полях. Указанные значения являются агрегированными внутренними данными действующего реестра.

Метод 1. Экспериментальные замеры ($n=5$, тёплый кеш). SQL-запрос выполнялся пять раз подряд в одной сессии, после чего фиксировалось чистое время его выполнения (execution time). Затем вычислялись среднее значение \bar{t} , стандартное отклонение s и 95%-ный доверительный интервал: $\pm 2,776 \cdot s / \sqrt{5}$. Замеры проводились при тёплом кеше (warm cache), то есть в ситуации, когда рабочие данные уже находятся в оперативном буфере СУБД после предшествующих обращений. Такой режим соответствует типичной непрерывной работе оператора. Для полного реестра (сценарий 0) приведён одиночный ориентировочный замер (\dagger), поэтому доверительный интервал для него не рассчитывался.

Метод 2. Анализ плана запроса (EXPLAIN ANALYZE). Этот метод использовался для получения фактического плана выполнения запроса с реальными значениями rows – числа строк, обработанных на каждом шаге, и loops – числа повторных запусков соответствующего шага – до и после оптимизации.

Метод 3. Оценка доверительного интервала для коэффициента ускорения методом бутстрэпа. Бутстрэп – непараметрический метод статистического оценивания, при котором из исходных данных многократно формируются псевдовыборки путём случайного выбора с возвращением. Распределение интересующей статистики по этим итерациям используется для вычисления доверительного интервала. Серия «до» была получена при $\text{nominal} = f_{\text{test}}$, серия «после» – при $\text{nominal} = f_{\text{oper}}$, то есть при операционном номинале

с ненулевым числом присвоений в регионе. Конкретные значения номиналов не раскрываются, поскольку для цели данного эксперимента существенны не предметные свойства частоты, а фактическая вычислительная нагрузка запроса. Эти два номинала не рассматриваются как полностью эквивалентные по предметному содержанию. Их сопоставление используется только для оценки изменения плана выполнения после создания индекса, поскольку EXPLAIN ANALYZE в обоих случаях показывает одинаковые ключевые параметры: соединение Owner \bowtie Site даёт 13 105 строк, а зависимый подзапрос выполняется 25 422 раза. Для оценки коэффициента ускорения было выполнено $B=100\,000$ псевдовыборок с возвращением по двум наборам замеров: до оптимизации {4172; 3137; 3134; 3113; 3038} мс и после оптимизации {557; 526; 510; 556; 490} мс. Для каждой итерации рассчитывалось отношение средних значений $\bar{t}_{\text{до}} / \bar{t}_{\text{после}}$; 95%-ный ДИ определялся по 2,5-му и 97,5-му перцентилям и составил 5,69–7,16 раза, медиана – 6,22 раза.

Метод 4. Подсчёт атрибутов схемы базы данных. Для приближённой структурной оценки подсчитывалось число атрибутов (DESCRIBE Owner/Site/Freq), которые оператор потенциально заполняет при ручном создании записи и при использовании операции Copy. Этот метод оценивает только сокращение числа полей ввода и не является хронометражем работы оператора или измерением частоты ошибок ввода. Ниже последовательно описаны механизмы АИС управления РЧС КР, ставшие объектом экспериментальной оценки.

Механизм многокритериального поиска

Механизм поиска построен на принципе последовательного сужения выборки: каждый новый критерий применяется не ко всему реестру заново, а к уже отфильтрованному множеству записей. Формально это можно описать так: пусть $P = \{p_1, \dots, p_n\}$ – полное множество доступных критериев, а $A \subseteq P$ – подмножество критериев, активированных оператором. Тогда условие выборки $W(A) = \bigwedge \{p_i | p_i \in A\}$ представляет собой логическое И всех активных условий, а результирующее множество $F'(A)$ не возрастает при расширении A . Практически это означает, что каждый новый критерий либо сокращает число кандидатов, либо оставляет его прежним, но не расширяет. Используемые критерии охватывают административные, пространственные, технические и регистрационные параметры трёх уровней (табл. 1) [1; 4].

Таблица 1

Критерии поиска в реестре РЧС КР

Ур.	Критерий	Поле БД	Тип сравнения
1	Идентификатор владельца	owner.ID	Точное =
1	Наименование организации	owner.Name	Подстрочный LIKE
1	Регион КР (9 кодов)	owner.area (int)	По перечислению
1	Статус лицензии	owner.state (int)	По перечислению
1	Номер счёта / лицензии	owner.schet / LICNUMnum	Точное =
2	Наименование объекта	site.SiteName	Подстрочный LIKE
2	Широта / долгота (°, ', ")	site.latitude/longitude 0/1/2	Точное = по компонентам
3	Номинал частоты (МГц → кГц)	freq.nominal (double)	Точное =, конверсия единиц
3	Полоса частот	freq.band	Точное =

Примечание: составлено авторами. Поле area: 9 кодов = 7 областей КР + Бишкек + Ош; поле state – целочисленные коды; номинал – в кГц.

Режим быстрого обзора byOne («по одной записи на объект») предназначен для компактного просмотра реестра: вместо вывода всех частотных присвоений система показывает только первое найденное присвоение для каждого объекта. Технически режим реализован через подзапрос SELECT f2.ID FROM Freq f2 WHERE f2.IDsite=s.ID LIMIT 1. В нормальной работе byOne используется без привязки к конкретной частоте; нагрузочные замеры с фильтром по номиналу проводились только для анализа плана выполнения. При активном фильтре по номиналу запрос может возвращать 0 строк, поскольку первое найденное присвоение не всегда совпадает с заданным значением. Однако число запусков подзапроса определяется количеством объектов в регионе, а не числом строк в итоговой выборке, поэтому данный режим использовался для анализа вычислительной нагрузки, а не пользовательской скорости просмотра.

Механизм каскадного копирования типовой записи

Механизм поиска прототипа служит отправной точкой для операции каскадного копирования. Операция работает с тремя уровнями данных: лицензиат, объект размещения радиоэлектронных средств и частотное присвоение, что соответствует общей логике хранения и выборки сведений в автоматизированных системах управления радиочастотным спектром [1]. Формально операция Copy(ow_id) порождает дерево $ow' \rightarrow \{s'\} \rightarrow \{f'_{ij}\}$: новый Owner наследует технические поля прототипа, каждый Site воспроизводится с новым IDowner, а каждая Freq – с новыми IDsite и IDowner. После создания каждой записи Freq вызы-

вается integrityService.logInsertById(), который вычисляет HMAC-SHA-256 для канонического представления данных записи и сохраняет результат в журнале целостности. В данной статье журналирование рассматривается как механизм фиксации создаваемых записей и контроля последующей целостности, но не как доказательство полной атомарности операции Copy при сбоях. Технически операция реализована как каскад JDBC-вставок; перевод всей процедуры Owner→Site→Freq в явную ACID-транзакцию рассматривается как задача следующей версии. Следовательно, безопасность Copy при частичных сбоях не является как достигнутый результат настоящего исследования.

Механизмы экспорта и подготовки печатных форм

Отобранные и скопированные записи могут передаваться во внешние форматы с помощью механизмов экспорта и подготовки печатных форм. Экспорт реализован как выгрузка фильтрованной выборки F' в формат XLSX без повторного ввода параметров запроса. Подсистема печати формирует PDF-документ после проверки целостности выбранных записей; при выявлении несоответствия операция блокируется. Для сформированного документа дополнительно вычисляется контрольная метка на основе SHA-256. В настоящей статье эти функции описываются только как элементы рабочего контура системы. Их влияние на производительность работы оператора количественно не измерялось, поэтому далее основное внимание уделено поиску, плану выполнения запроса и эффекту индексирования.

Результаты исследования и их обсуждение

Производительность механизма поиска представлена в таблице 2. Последовательное добавление предикатов монотонно сокращает выборку. Нулевое число записей в сценариях 4–5 не является ошибкой: оно показывает, что для данной комбинации условий в реестре просто нет совпадений. В этом смысле нулевой результат тоже является содержательно значимым, поскольку демонстрирует предельную избирательность механизма поиска.

Диагностика режима byOne и эксперимент по оптимизации. До оптимизации среднее время выполнения составляло $\bar{t}=3\,318,8$ мс (95%-ный ДИ: $\pm 594,2$ мс). Сокращённый план EXPLAIN ANALYZE до оптимизации показал три ключевых признака: полное сканирование таблицы Site (271 453 строки), формирование соединения Owner \bowtie Site объёмом 13 105 строк и 25 422 итерации зависимого подзапроса.

Причина задержки связана с порядком доступа к данным. Из-за отсутствия индекса на Owner.area MySQL начинает выполнение с полного сканирования Site (271 453 строки), затем для каждой строки Site обращается к Owner и проверяет условие area=2; в результате соединение Owner \bowtie Site даёт 13 105 строк. Зависимый подзапрос запускается 25 422 раза, что зафиксировано в поле loops плана EXPLAIN ANALYZE. В литературе по коррелированным подзапросам отмечается, что та-

кая стратегия может приводить к повторному выполнению подзапроса для кортежей родительского запроса [14], а исследования сложных запросов показывают зависимость времени выполнения от структуры запроса, индексирования и оптимизации [15]. В настоящей работе конкретный эффект индекса Owner.area определяется фактическим планом EXPLAIN ANALYZE и собственными замерами.

Репрезентативность замеров и эксперимент после оптимизации. После создания индекса (CREATE INDEX idx_owner_area ON Owner(area)) режим byOne был протестирован на сценарии с nominal=f_oper – операционным номиналом, для которого в Чуйском регионе зарегистрировано $O(10^2)$ объектов. Это подтверждено запросом COUNT(DISTINCT Site.ID); конкретное значение f_oper не публикуется. Нулевое число строк в результате byOne объясняется тем, что первая запись Freq по LIMIT 1 для конкретного объекта может не совпасть с запрошенным номиналом. Однако EXPLAIN ANALYZE показывает, что соединение Owner \bowtie Site по-прежнему даёт 13 105 строк, а подзапрос выполняется 25 422 раза. Следовательно, измерение отражает объём вычислительной работы данного плана запроса. При этом nominal=f_test и nominal=f_oper не трактуются как полностью сопоставимые по предметному содержанию; их сопоставимость ограничена совпадением ключевых характеристик фактического плана выполнения.

Таблица 2

Производительность механизма поиска
(MySQL 8.0.45, n=5, тёплый кеш)

Сценарий (активные предикаты)	Замеры t, мс	\bar{t} , мс	s, мс	95% ДИ	Записей
Полный реестр (ориент. †)	2 584†	-	-	-	667 193
area=Чуйская (код 2)	53; 53; 52; 70; 52	56,0	7,8	$\pm 9,7$	24 564
area + state=Действующее (код 2)	34; 40; 28; 33; 51	37,2	8,8	$\pm 10,9$	15 047
area + nominal=f_test	17; 11; 11; 17; 10	13,2	3,5	$\pm 4,3$	0
area + state + nominal=f_test	18; 10; 10; 9; 16	12,6	4,1	$\pm 5,1$	0
byOne (до оптимизации, nom=f_test)	4 172; 3 137; 3 134; 3 113; 3 038	3 318,8	478,6	$\pm 594,2$	0
byOne (после INDEX, nom=f_oper)	557; 526; 510; 556; 490	527,8	29,1	$\pm 36,2$	0*

Примечание: составлено авторами. (†) – одиночный ориентировочный замер, ДИ не вычислялся. (*) – 0 строк в результате: режим byOne выбирает первую запись Freq по LIMIT 1, которая может не совпадать с запрошенным номиналом; это не патологический случай – EXPLAIN ANALYZE подтверждает одинаковое число итераций независимо от результата (см. текст). Число объектов с nominal=f_oper в регионе: $O(10^2)$; конкретные значения f_test и f_oper не публикуются, поскольку в данном сравнении существенны не предметные свойства частот, а совпадение ключевых характеристик фактического плана выполнения запроса. $95\% \text{ ДИ} = \pm 2,776 \cdot s / \sqrt{5}$.

Таблица 3

Планы EXPLAIN ANALYZE для режима byOne до и после оптимизации

Характеристика плана	До CREATE INDEX	После CREATE INDEX idx_owner_area
Точка входа	Full scan: Site (271 453 стр.)	Index lookup: Owner.idx_owner_area (4 654 стр.)
Строк соединения Owner ⋈ Site (actual)	13 105	13 105 (без изменений)
Итераций подзапроса (loops, EXPLAIN ANALYZE)	25 422	25 422 (без изменений)
Сценарий замеров	nominal=f_test	nominal=f_oper (O(10 ²) объектов)
\bar{t} , мс (95% ДИ)	3 318,8 (±594,2)	527,8 (±36,2)
Ускорение (95%-й ДИ, бутстрэп)	-	в 6,2 раза [5,69–7,16 раза]

Источник: расчёты авторов по фактическим значениям EXPLAIN ANALYZE (actual rows, loops) и замерам времени выполнения запроса; бутстрэп: B=100 000 псевдовыборок, исходные замеры (табл. 2). Данные отражают внутренний агрегированный набор действующего реестра; подтверждающие неперсонализированные материалы могут быть предоставлены редакции по запросу.

После оптимизации среднее время выполнения составило $\bar{t}=527,8$ мс ($s=29,1$ мс, 95%-ный ДИ: $\pm 36,2$ мс). Для коэффициента ускорения, рассчитанного методом бутстрэпа, 95%-ный доверительный интервал составил 5,69–7,16 раза. EXPLAIN ANALYZE показал изменение точки входа в запрос: вместо полного сканирования Site (271 453 строки) использован индексный поиск по Owner.idx_owner_area (4 654 строки). При этом число итераций зависимого подзапроса (25 422) и число строк соединения Owner ⋈ Site (13 105) не меняются, то есть сам подзапрос оптимизацией не затронут. Следовательно, ускорение достигается именно за счёт устранения полного сканирования промежуточной таблицы. Сводное сопоставление планов до и после оптимизации приведено в таблице 3.

Обобщаемость выявленной закономерности. Полученный результат не следует трактовать как доказательство универсального ускорения во всех реестровых системах. Он описывает конкретную закономерность плана выполнения запроса, выявленную в исследованном реестре. Исследования производительности сложных запросов показывают, что эффективность выполнения зависит от структуры запроса, сложности связей между данными, индексирования и оптимизации [15]. Поэтому аналогичный эффект может ожидать только при одновременном выполнении трёх условий: (а) на верхнем уровне есть категориальный атрибут с низкой кардинальностью, используемый как основной фильтр; (б) запрос к нижнему уровню реализован как зависимый подзапрос; (в) индекс на этом атрибуте отсутствует. В такой ситу-

ации добавление индекса может изменить точку входа в план запроса и устранить полное сканирование промежуточной таблицы. Смежные публикации по индексированию и оптимизации выполнения SQL-запросов подтверждают, что выбранная стратегия доступа к данным и способ исполнения запроса могут существенно влиять на план и время выполнения [16; 17]; при этом конкретная величина ускорения в 6,2 раза относится только к исследованному набору данных и измеренному запросу.

Оценка числа потенциально заполняемых полей. Подсчёт атрибутов показывает, что при ручном создании записи оператор потенциально заполняет около 50 полей (Owner + Site + Freq), тогда как операция Сору оставляет для заполнения около 15 административных полей Owner. Таким образом, структурное сокращение числа потенциально заполняемых полей составляет примерно 70%. Этот показатель не следует интерпретировать как равное снижение времени работы оператора или как доказанное снижение числа ошибок ввода, поскольку хронометраж действий пользователей и анализ ошибок в настоящем исследовании не проводились. Архивные данные журнала предшествующей системы показывают, что в 2008–2012 гг. функция копирования использовалась в среднем 2 061 раз в год; эти данные подтверждают практическую востребованность сценария копирования, но не являются прямым измерением его текущего временного эффекта.

Интегрированный рабочий контур

Полученные результаты позволяют описать поиск и каскадное копирование как

связанные элементы рабочего контура. Сначала оператор выполняет поиск: в измеренных сценариях типовые запросы занимали 7–56 мс и постепенно сужали выборку от 667 193 записей до нуля. Затем может выполняться копирование: система воспроизводит дерево Owner→Site→Freq, а число потенциально заполняемых оператором полей сокращается примерно с 50 до 15. Такая логика повторного использования уже имеющихся сведений концептуально согласуется с принципом однократного предоставления данных [9; 10], однако в данной статье проверяется не общий административный эффект, а конкретные технические показатели поиска, индексирования и структурного сокращения числа полей ввода. Далее возможны экспорт выборки F' в XLSX и формирование PDF после проверки целостности выбранных записей и вычисления контрольной метки документа. Экспорт и печать в настоящей статье описываются только функционально: их влияние на производительность рабочего процесса количественно не оценивалось.

Ограничения исследования

При интерпретации результатов важно учитывать несколько ограничений. Во-первых, замеры проводились при $n=5$ и тёплом кеше, поэтому при холодном кеше численные значения могут отличаться. Во-вторых, подсчёт атрибутов схемы даёт только структурную оценку сокращения числа потенциально заполняемых полей и не доказывает пропорциональное снижение времени работы оператора. В-третьих, в исследовании не проводился хронометраж действий пользователей и не измерялась частота ошибок ввода до и после применения `Сору`. В-четвёртых, данные об интенсивности операций копирования (2 061 в год) относятся к предшествующей системе и не обязательно полностью отражают текущую практику. В-пятых, сравнение $\text{nominal}=f_{\text{test}}$ и $\text{nominal}=f_{\text{oper}}$ ограничено совпадением ключевых характеристик фактического плана выполнения запроса и не означает полной предметной эквивалентности этих частотных номиналов. В-шестых, сопоставление с близкими по тематике публикациями [11; 12] опирается на ограниченный массив открыто доступных работ и не претендует на исчерпывающий обзор всех специализированных систем управления спектром. В-седьмых, количественная оценка производительности фаз экспорта и подготовки печатных форм в данной статье не проводилась. Наконец, операция `Сору` в текущей реализации описана как каскад JDBC-вставок; её

оформление как явной ACID-транзакции относится к дальнейшей доработке.

Выводы

С учётом полученных результатов, их обсуждения и обозначенных ограничений можно сформулировать основные выводы. В статье описаны и экспериментально оценены механизмы поиска прототипа и каскадного копирования в государственном радиочастотном реестре. Ключевые результаты таковы.

1. Типовые запросы поиска выполняются за 7–56 мс; последовательное добавление предикатов сокращает пространство кандидатов от 667 193 до 0 записей.

2. Посредством EXPLAIN ANALYZE диагностирована закономерность: отсутствие индекса на Owner.area (9 значений) → полный скан Site (271 453 строки) → $f=3\ 318,8$ мс. Создание INDEX(area) даёт $f=527,8$ мс (ускорение в 6,2 раза; 95%-ный ДИ, бутстрэп: 5,69–7,16 раза), что подтверждено на операционном сценарии с ненулевым числом объектов для $\text{nominal}=f_{\text{oper}}$. Сопоставление сценариев основано на совпадении ключевых характеристик фактического плана выполнения запроса, а не на полной предметной эквивалентности $\text{nominal}=f_{\text{test}}$ и $\text{nominal}=f_{\text{oper}}$. За пределами исследованного реестра результат следует рассматривать как архитектурно обусловленное ожидание, применимое только при наличии сходного сочетания условий: категориального фильтра верхнего уровня, зависимого подзапроса и отсутствующего индекса по полю фильтрации.

3. Подсчёт атрибутов показывает снижение числа потенциально заполняемых оператором полей с около 50 до около 15, то есть примерно на 70%. Этот результат характеризует структурное сокращение объёма ввода, но не является прямым измерением времени работы оператора или частоты ошибок ввода.

К направлениям дальнейших исследований относятся хронометражное исследование с участием операторов, измерение частоты ошибок ввода до и после применения `Сору`, оформление операции `Сору` как явной ACID-транзакции, количественная оценка экспорта и печати, а также расширение эксперимента ($n \geq 30$, холодный кеш).

Список литературы

1. ITU-R. Recommendation ITU-R SM.1370-2. Design guidelines for developing automated spectrum management systems [Электронный ресурс]. URL: <https://www.itu.int/rec/R-REC-SM.1370-2-201308-1/en> (дата обращения: 22.03.2026).
2. ITU-R. Справочник по управлению использованием спектра на национальном уровне. Женева: Международный союз электросвязи, 2005. [Электронный ресурс]. URL:

https://www.itu.int/dms_pub/itu-r/opb/hdb/R-HDB-21-2005-PDF-R.pdf (дата обращения: 22.03.2026).

3. ITU-R. Handbook on Computer-Aided Techniques for Spectrum Management (CAT). Edition of 2015 // ITU. [Электронный ресурс]. URL: https://www.itu.int/dms_pub/itu-r/opb/hdb/R-HDB-01-2015-PDF-E.pdf (дата обращения: 22.03.2026).

4. ITU-R. Руководство по регламентарной структуре для управления использованием спектра на национальном уровне [Электронный ресурс]. Женева: Международный союз электросвязи, 2010. URL: <https://www.itu.int/pub/R-REP-SM.2093-1-2010> (дата обращения: 22.03.2026).

5. Вартамян Ф. А. Электронное правительство как инструмент повышения качества государственных услуг // Вестник Евразийской науки. 2025. Т. 17. № s2. URL: <https://esj.today/89favn225.html> (дата обращения: 14.03.2026).

6. Gulyukin D. O. Improving mechanisms for digital transformation of public administration at the regional level: problems and prospects // Сибирский журнал экономических и бизнес-исследований. 2025. Т. 14. № 4. С. 63–90. DOI: 10.12731/3033-5973-2025-14-4-321. URL: <https://kras-science.ru/jour/index.php/nk/article/view/321> (дата обращения: 14.03.2026).

7. Богдановский И. Ю. Цифровизация государственных услуг: моделирование и мониторинг эффективности // Путеводитель предпринимателя. 2025. Т. 18. № 1. С. 9–14. DOI: 10.24182/2073-9885-2025-18-1-9-14. URL: <https://www.pp-mag.ru/jour/article/view/2059> (дата обращения: 14.03.2026).

8. Малик Е. Н., Шедий М. В., Пикалов Б. В. «Электронное правительство» как интегративная платформа предоставления государственных услуг // Управленческое консультирование. 2020. № 9. С. 19–30. DOI: 10.22394/1726-1139-2020-9-19-30. URL: <https://www.acjournal.ru/jour/article/view/1511> (дата обращения: 14.03.2026).

9. Wimmer M. A. Once-Only Principle Good Practices in Europe // The Once-Only Principle: The TOOP Project / edited by Krimmer R., Prentza A., Mamrot S. Cham: Springer International Publishing, 2021. С. 61–82. DOI: 10.1007/978-3-030-79851-2_4. ISBN: 978-3-030-79851-2.

10. Wimmer M. A., Neuron A. C., Frecè J. T. Approaches to Good Data Governance in Support of Public Sector Transformation Through Once-Only // Electronic Government / edited by Viale Pereira G. and others. Cham: Springer, 2020. С. 210–222. DOI: 10.1007/978-3-030-57599-1_16. URL: https://link.springer.com/chapter/10.1007/978-3-030-57599-1_16 (дата обращения: 14.03.2026).

[er.com/chapter/10.1007/978-3-030-57599-1_16](https://link.springer.com/chapter/10.1007/978-3-030-57599-1_16) (дата обращения: 14.03.2026). ISBN: 978-3-030-57599-1.

11. Наговицин А. И., Молоткова Б. Б., Азимов Д. У. Автоматизированная система контроля за назначением радиочастот на основе риск-ориентированного подхода при планировании радиосвязи в подразделениях робототехнических комплексов военного назначения // Известия ЮФУ. Технические науки. 2022. № 1. С. 59–70. DOI: 10.18522/2311-3103-2022-1-59-70. URL: https://izv-tn.tti.sfedu.ru/index.php/izv_tn/article/view/596 (дата обращения: 14.03.2026).

12. Головской В. А. Функциональная модель подсистемы управления ресурсами когнитивной радиосистемы робототехнического комплекса // Известия ЮФУ. Технические науки. 2023. № 1. С. 241–251. DOI: 10.18522/2311-3103-2023-1-241-251. URL: https://izv-tn.tti.sfedu.ru/index.php/izv_tn/article/view/761 (дата обращения: 14.03.2026).

13. Аданбаев А. М., Шантаев Н. О. Система управления радиочастотами: свидетельство об официальной регистрации программы для ЭВМ № 964. Киргизпатент, 2025. URL: https://base.patent.kg/evm.php?action=search_list&number_svid=964 (дата обращения: 22.03.2026).

14. Zhang C., Liu W., Pang T., Yue Y. Optimization of correlate subquery based on distributed database // Journal of Northwestern Polytechnical University. 2021. Vol. 39. № 4. P. 909–918. DOI: 10.1051/jnwpu/20213940909. URL: <https://www.jnwpu.org/articles/jnwpu/abs/2021/04/jnwpu2021394p909/jnwpu2021394p909.html> (дата обращения: 14.03.2026).

15. Kotiranta P., Junkkari M., Nummenmaa J. Performance of Graph and Relational Databases in Complex Queries // Applied Sciences. 2022. Vol. 12. № 13. P. 6490. DOI: 10.3390/app12136490. URL: <https://www.mdpi.com/2076-3417/12/13/6490> (дата обращения: 14.03.2026).

16. Abbasi M., Bernardo M. V., Váz P., Silva J., Martins P. Optimizing Database Performance in Complex Event Processing through Indexing Strategies // Data. MDPI. 2024. Vol. 9. № 8. P. 93. DOI: 10.3390/data9080093. URL: <https://www.mdpi.com/2306-5729/9/8/93> (дата обращения: 14.03.2026).

17. Пантелимонов М. В., Бучачский Р. А., Жуйков Р. А. Кэширование машинного кода в динамическом компиляторе SQL-запросов для СУБД PostgreSQL // Труды Института системного программирования РАН. 2020. Т. 32. № 1. С. 205–220. DOI: 10.15514/ISPRAS-2020-32(1)-11. URL: <https://ispranproceedings.elpub.ru/jour/article/view/1245> (дата обращения: 14.03.2026).

Конфликт интересов: Авторы заявляют о наличии следующих конфликтов интересов: Аданбаев А. М. является разработчиком программного средства «Система управления радиочастотами», использованного в исследовании и официально зарегистрированного в Киргизпатенте (свидетельство № 964). Автор Сагымбаев А. А. заявляет об отсутствии конфликта интересов.

Conflict of interest: The authors state that there are the following conflicts of interest: A. M. Adanbayev is the developer of the Radio Frequency Management System software used in the study and officially registered with Kyrgyzpatent (certificate № 964). The author Sagymbaev A. A. declares that there is no conflict of interest.

Финансирование: Авторы заявляют об отсутствии внешнего финансирования.

Financing: The research was performed without external funding.