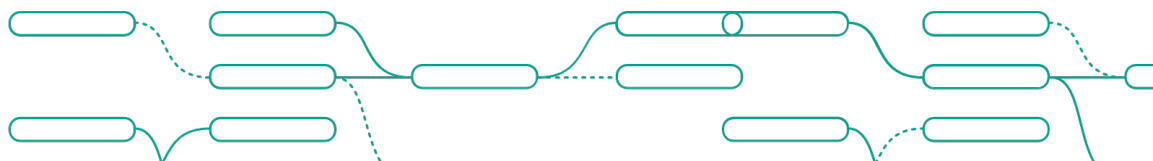


Arenadata Catalog Data Quality Framework

Функциональные возможности ADC.DQF

Москва 2026



Оглавление

Введение	3
Термины и определения	3
Сокращения и обозначения	3
Общие положения	3
Возможности	4
Особенности	5
Редактор правил	6
Настройка правил	6
Базовый алгоритм	6
Механизм управления потоком	9
Правило	10
Проверка корректности правила	11
Меню редактора правил	11
Каталог правил	13
Список правил	13
Список групп	13
Создание группы	14
Связь группы правил с тест-кейсами Arenadata Catalog	15
Планирование задач	17
Список задач	17
Создание задачи	17
Изменение статуса задачи	18
Просмотр результатов выполнения задачи	19
Отчетность	21
Формирование и загрузка отчетов	22
Дашборды	23
Добавление и изменение виджетов	24
Редактирование дашборда	26
Сценарии взаимодействия с приложением	26
Владелец процесса качества данных	27
Аналитик качества данных	27
Владелец источника данных	27
Потребитель данных	28

Введение

Термины и определения

Термин	Значение
Базовый алгоритм	Минимальная логическая конструкция, определяющая основную операцию проверки, получения или трансформации данных.
Механизм управления потоком	Связующее звено, объединяющее базовые алгоритмы в единую последовательность исполнения – правило.
Правило	Структурированная последовательность шагов (базовых алгоритмов, механизмов управления потоком и других правил), где каждый шаг возвращает статус выполнения и оперирует данными через контекст исполнения для решения конкретной бизнес-задачи.
Проверка	Процесс исполнения заданного правила над исходными данными для выявления несоответствий или подтверждения их качества.
Группа	Совокупность правил, объединенных вместе в соответствии с бизнес-потребностью.
Задача	Настраиваемый объект, запускающий выполнение группы правил на указанном наборе данных. Включает множество проверок, каждая из которых представляет собой применение одного правила к одному объекту данных.
Дашборд	Интерактивная панель с графиками и ключевыми метриками
JSON / JSON-тип	Это текстовый формат обмена данными, который используется для структурированного представления данных.
JSONPath	Синтаксис для обращения к элементам внутри структуры данных в формате JSON.
YAML	Формат сериализации данных, разработанный для удобства восприятия человеком и предназначенный для представления структурированных данных, особенно в файлах конфигурации
Drag-and-drop	Метод взаимодействия с элементами пользовательского интерфейса, при котором пользователь выбирает объект, перемещает его, удерживая нажатой кнопку мыши или касаясь экрана, и затем отпускает его в нужном месте
Drill-down	Аналитический подход, при котором пользователь постепенно переходит от общей информации к более детальной, изучая данные на разных уровнях иерархии

Сокращения и обозначения

Сокращение	Наименование
ADC DQF	Arenadata Catalog Data Quality Framework
СУБД	Система управления базами данных
DSL	Предметно-ориентированный язык (язык программирования, специализированный для конкретной области применения)

Общие положения

Настоящий документ является кратким описанием функциональных возможностей программного обеспечения Arenadata Catalog Data Quality Framework (ADC DQF). ADC DQF

– это программное обеспечение, позволяющее производить верификацию данных с использованием настраиваемых правил, сконфигурированных в соответствии с особенностями предметной области и бизнес-логики заказчика. ADC DQF может быть интегрирован в любую среду оркестрации и обеспечивать проверку записей в памяти.

Возможности

Выполнение проверок данных по заданным алгоритмам и настраиваемым параметрам

ADC DQF выполняет проверки данных с использованием гибких алгоритмов, которые можно адаптировать под специфические потребности бизнеса в соответствии с его уникальными процессами и условиями. Настройка алгоритмов позволяет анализировать данные в зависимости от их типа и значимости, обеспечивая гибкость и точность при проверке данных разных категорий и уровней. Например, конфигурационные возможности ПО позволяют задать параметры для проверки на основании заранее определённых правил, таких как: логические условия, диапазоны допустимых значений, соответствие определённым форматам (даты, числа, символы), корректность взаимосвязей между данными.

Выполнение проверок по массивам данных и единичных записей

DQF поддерживает проверку как больших объёмов данных, так и отдельных записей. Это позволяет выявлять ошибки и несоответствия на уровне как всей базы данных, так и отдельных элементов, обеспечивая высокую точность контроля качества данных. Например, ежемесячная проверка всех записей в базе данных конкретного отдела или проверка единичной записи, когда она впервые заносится в информационную систему.

Выявление ошибок (противоречий) в имеющихся и вносимых данных

DQF обнаруживает ошибки и противоречия как в уже существующих данных, так и в новых данных, которые вносятся в систему. Это позволяет предотвратить накопление ошибок и улучшить общую точность и согласованность данных. Например, если новая запись конфликтует с уже имеющимися данными (противоречит формату, структуре и т.д.), ПО идентифицирует это несоответствие и возвращает необходимую в этом случае ошибку.

Верификация невалидных данных с возможностью их замены

DQF может проверять невалидные данные и предлагать их замену корректными значениями. Этот процесс может быть выполнен вручную или автоматически посредством настраиваемого расписания, что значительно ускоряет корректировку ошибок и позволяет поддерживать данные в актуальном состоянии. Например, ПО может указать на неправильный формат или предложить стандартное значение из внешнего справочника в случае получения пустых полей.

Постоянный мониторинг и оценка качества данных на предмет полноты, достоверности и непротиворечивости

DQF постоянно отслеживает качество данных, оценивая их по ключевым показателям — полноте, достоверности и непротиворечивости. Этот мониторинг помогает своевременно выявлять проблемы и поддерживать высокий уровень качества данных. Например, система может проверять, заполнены ли все обязательные поля в каждой записи, соответствует ли формат данных установленным требованиям, а также согласуются ли данные между собой (проверка на дублирующие или противоречивые записи).

Особенности

В DQF реализована возможность работать с данными в широком смысле этого слова. ПО поддерживает работу с различными классификациями данных вне зависимости от их структуры, типа и отраслевой специфики, поскольку оперирует с ними, как с абстрактными объектами за счет использования настраиваемых алгоритмов проверок. Это значит, что система не привязана к конкретной предметной области и может применяться для самых разных типов данных. Благодаря гибкой настройке, проверки данных можно адаптировать под любую сферу, просто изменив конфигурацию правил и алгоритмов — без необходимости привлекать разработчиков для внесения изменений.

При верификации данных DQF осуществляет контроль за ходом выполнения проверок данных, после чего предоставляет результаты в зависимости от конфигурации, используемого при проверке алгоритма. При необходимости DQF может осуществлять запросы дополнительных данных, необходимых для проведения конкретной проверки, в том числе из внешних информационных систем.

DQF осуществляет автоматизированные проверки по трем направлениям:

1. Форматно-логический контроль:
 - наличие обязательных атрибутов;
 - соответствие атрибутов заданной длине и маске;
 - соответствие атрибутов требуемым форматам – буквенно-числовые последовательности, непечатаемые символы, специальные символы;
2. Проверки внутри модели данных:
 - проверки по внутренним справочникам;
 - проверки на соответствие нормативно-правовым актам;
 - проверки на связность и непротиворечивость объектов.
3. Интеграционные проверки с использованием внешних источников:
 - проверки по внешним справочникам;
 - проверки по данным из внешних информационных систем.

Редактор правил

Редактор правил — это визуальный инструмент для создания, просмотра и редактирования правил, описывающий логику обработки и анализа данных (Рисунок 1). Он представляет графический интерфейс, позволяющий пользователям без навыков программирования легко формировать логические условия, определяющие, каким требованиям должны соответствовать данные.

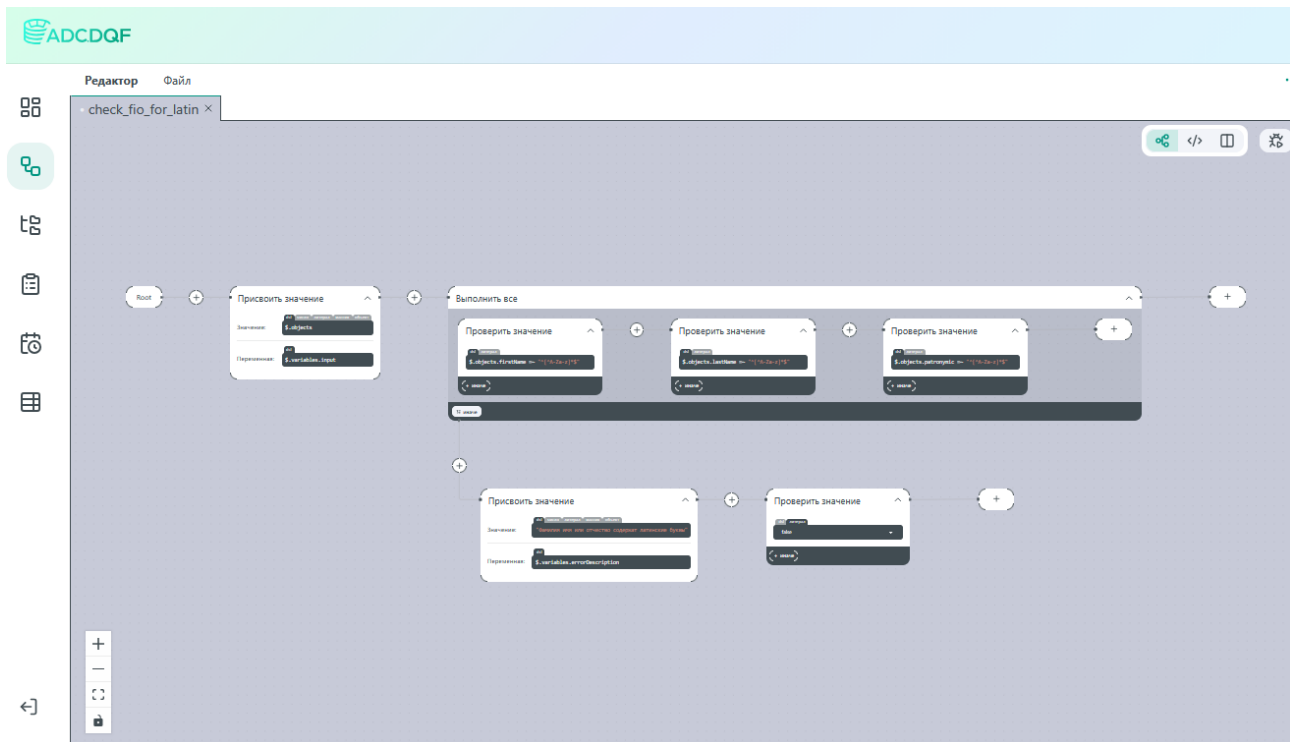


Рисунок 1 - Редактор правил

Настройка правил

Правило – структурированная последовательность шагов (базовых алгоритмов, механизмов управления потоком и других правил), где каждый шаг возвращает статус выполнения и оперирует данными через контекст исполнения для решения конкретной бизнес-задачи.

Пользователь может формировать структуру правила, добавляя или удаляя отдельные блоки. Каждый блок представляет собой либо базовый алгоритм, либо механизм управления потоком, либо вызывает выполнение другого сконфигурированного правила.

Базовый алгоритм

Базовый алгоритм - минимальная логическая конструкция, определяющая основную операцию проверки, получения или трансформации данных.

Для понимания возможностей проверки данных ниже (Таблица 1) приведена таблица с перечнем доступных базовых алгоритмов, которые могут быть включены в правило.

Название	Идентификатор	Описание
----------	---------------	----------

Проверить значение	CheckValue	Проверка и сравнение значений атрибутов
Присвоить значение	SetValue	Установка значения атрибута
Преобразовать массив	Collect	Сбор и фильтрация данных
Добавить в массив	AppendValue	Добавление атрибута в список
Продолжить	Continue	Продолжить исполнение
Содержится в справочнике	CheckValueFromDictionary	Проверка значения на вхождение в справочник
GraphQL запрос	GraphQLQuery	Запрос связанных объектов из GraphQL-сервиса
HTTP запрос	HttpRequest	Вызов HTTP методов
JDBC запрос	JDBCRequest	Выполнение SQL-запроса к поддерживаемым СУБД через стандартный интерфейс JDBC.
Проверить СНИЛС	CheckSnils	Проверка СНИЛС подсчетом контрольного числа в соответствии с алгоритмом, присвоения маркера качества
Проверить ИНН физического или юридического лица	CheckInn	Проверка идентификационного номера налогоплательщика (далее – ИНН) подсчетом контрольного числа в соответствии с алгоритмом для физических или юридических лиц (зависит от конфигурации базового алгоритма), присвоение маркера качества
Проверить ОГРН	CheckOgrn	Проверка и очистка государственного регистрационного номера записи о создании юридического лица либо записи о первом представлении сведений о юридическом лице (ОГРН)
Проверить адрес ¹	CheckAddress	Проверка адреса на вхождение в Государственный адресный реестр (далее – ГАР)

Таблица 1 - Список базовых алгоритмов

Ниже представлен пример вызова базового алгоритма "Добавить в массив" (AppendValue) в yaml-формате, который помещает элемент из пути \$.variables.recordNumber в массив по пути \$.variables.recordNumbersWithErrors.

```
AppendValue:
```

¹ Требуется подключение продукта «Гражданский Фактор» ООО «Клин Дейта» или «Фактор» ООО «ХФ Лабс».

```
expr: $.variables.recordNumber
result: $.variables.recordNumbersWithErrors
```

YAML-описание выше соответствует следующему визуальному элементу в редакторе (Рисунок 2).

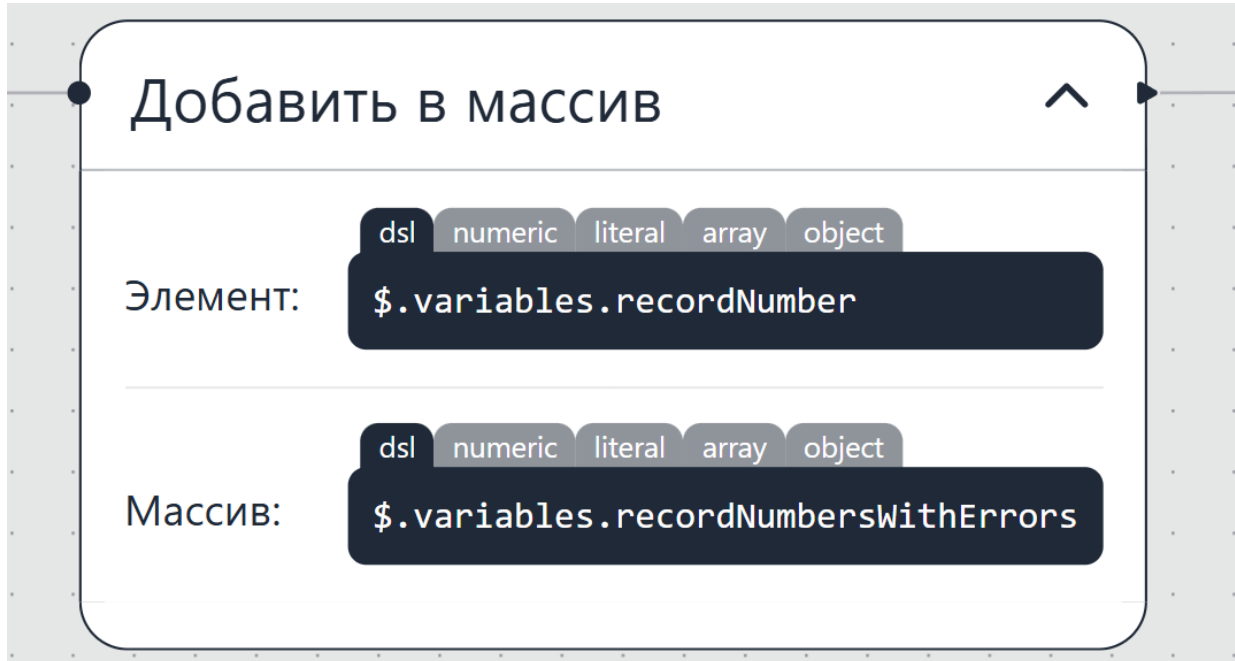


Рисунок 2 - Базовый алгоритм "Добавить в массив"

Типы полей

В визуальном представлении правила доступен выбор типа поля из набора predetermined значений (Таблица 2).

Тип поля	Наименование	Описание
dsl	Выражение	В выражении могут использоваться: <ul style="list-style-type: none"> • Константы любых типов, поддерживаемых в формате JSON — строки, числа, логические значения, null, объекты и массивы; • JSONPath-селекторы, позволяющие обращаться как к данным, переданным на вход, так и к данным, заданным внутри алгоритма; • Функции и операторы, с помощью которых можно проводить вычисления, задавать условия или обрабатывать данные. Такие конструкции могут быть вложенными и использовать другие выражения.
array	Массив	Упорядоченный список значений.
object	Объект	Набор пар «ключ: значение».
numeric	Число	Целое или дробное значение.
literal	Литерал	Специальные значения: true, false, null.

Таблица 2 - Типы полей

Отдельного внимания заслуживает тип «выражение» (dsl), используемый для вычисляемых условий. В выражениях можно использовать множество операторов и функций, поддерживаемых DQF.

Полный список доступных операторов и функций представлен в сопроводительной документации.

Механизм управления потоком

Механизм управления потоком — это связующее звено, объединяющее базовые алгоритмы в единую последовательность исполнения – правило.

Ниже перечислены доступные механизмы управления потоком, которые могут использоваться при настройке правила.

Название	Идентификатор	Описание
Выполнить все	all-of	Объединение последовательного исполнения шагов.
Повторять для	for-each	Итерирование по коллекции.
Повторять пока	while	Цикл, пока выполняется условие.
Иначе	otherwise	Альтернативная ветвь исполнения. <i>Не добавляется как отдельный самостоятельный блок, а прикрепляется к существующему элементу в редакторе, у которого предусмотрена логика разветвления.</i>

Таблица 3 - Механизмы управления потоком

Ниже (Блок кода 1) представлен пример вызова механизма управления потоком «Повторять пока» (while), внутри которого происходит вызов базового алгоритма «Присвоить значение» (SetValue).

```
do-at-least-once:
  SetValue:
    expr: $.variables.x - $.variables.n
    result: $.variables.x
  while: $.variables.x > 0
```

YAML-описание выше соответствует следующему визуальному элементу в редакторе (Рисунок 3).

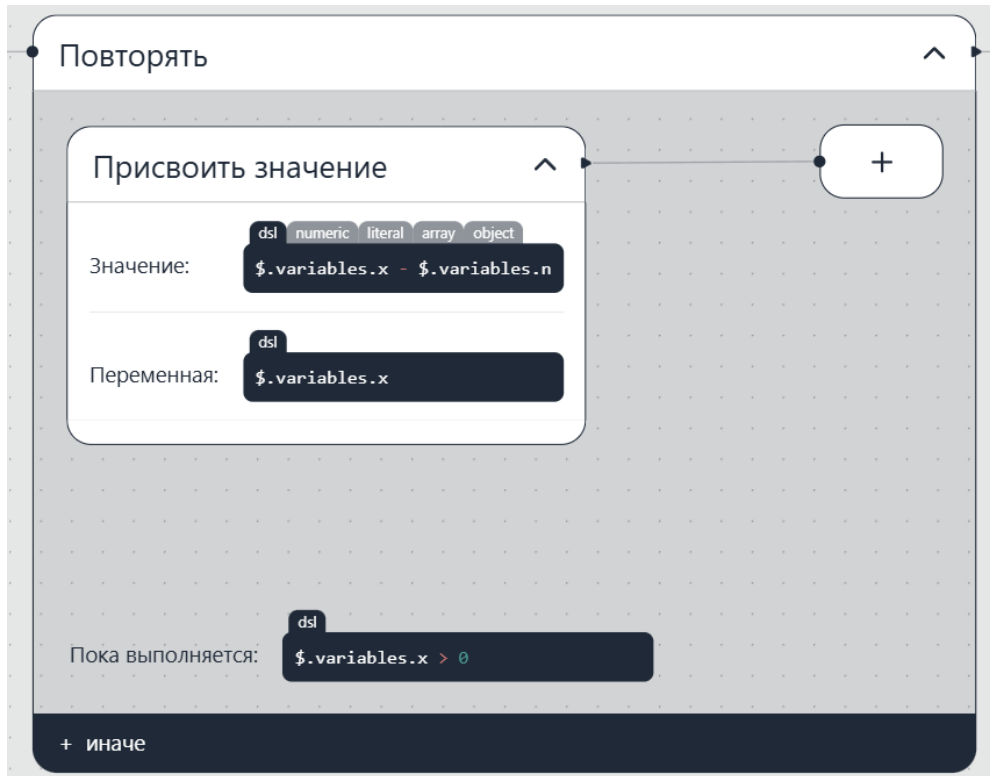


Рисунок 3 - Вызов механизма управления потоком «Повторять пока» с базовым алгоритмом "Присвоить значение"

Правило

При конфигурации правила, помимо выбора базовых алгоритмов и механизмов управления потоком, можно также использовать вызов другого правила. В этом случае во визуальном редакторе правило выбирается из каталога, а в текстовом формате необходимо указать его идентификатор вручную.

Ниже приведен пример вызова правила с идентификатором `bank_metrics`: сначала в формате YAML, затем его визуальное представление в редакторе (Рисунок 4).

```
- sample_algo: {}
```

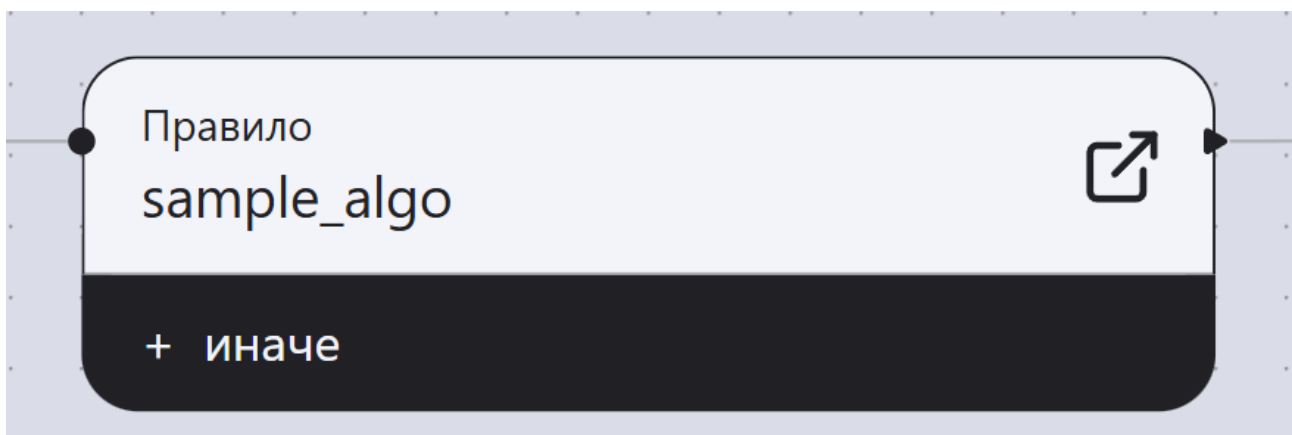


Рисунок 4 - Вызов правила sample_algo

Проверка корректности правила

Редактор правил предоставляет возможность протестировать правило непосредственно в интерфейсе (Рисунок 5). Пользователь может вручную задать входные данные и выполнить проверку, чтобы увидеть результат выполнения правила — включая итоговый статус и возвращаемые данные (Рисунок 6). Это особенно полезно на этапе разработки и отладки правил, позволяя оперативно выявить ошибки конфигурации или логики.

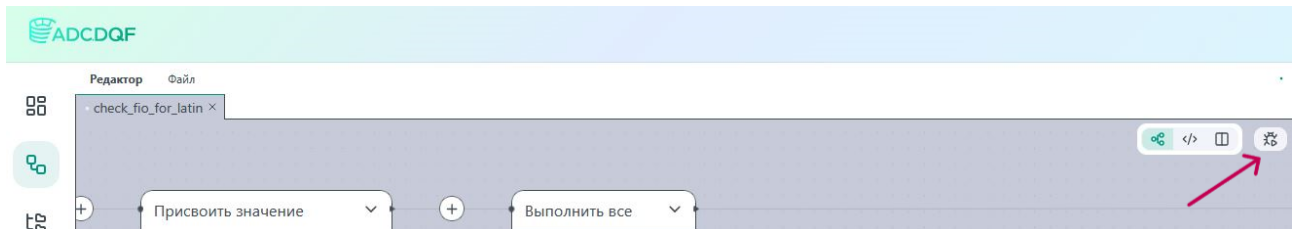


Рисунок 5 – Переход к проверке правила на тестовых данных

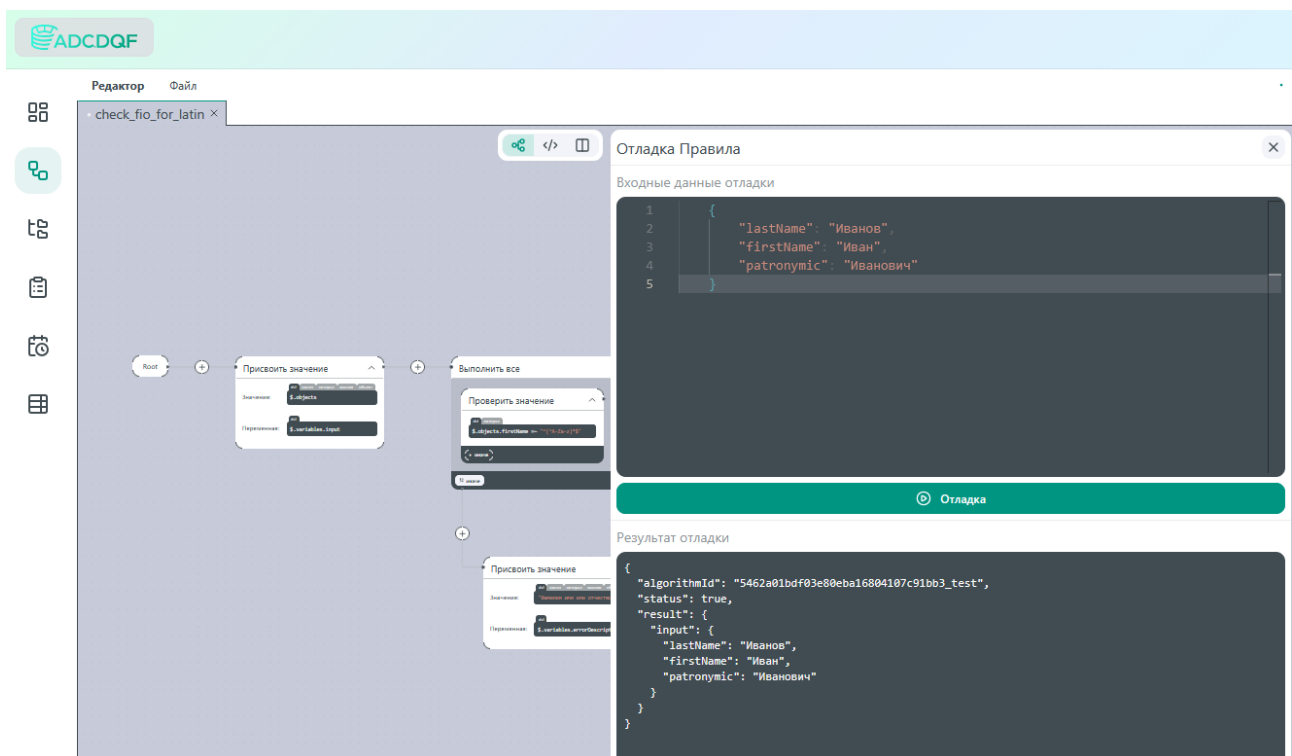


Рисунок 6 - Результат выполнения тестового запуска

Меню редактора правил

Меню редактора предоставляет доступ к основным действиям, связанным с управлением правилами (Рисунок 7). Через него пользователь может выполнять следующие операции:

- Открыть правило — выбрать любое существующее правило из каталога и загрузить его в редактор для просмотра или изменения;
- Создать новое правило — начать конфигурацию нового правила в новой вкладке редактора;
- Сохранить — сохранить изменения, внесённые в текущее правило;
- Сохранить как — сохранить правило под другим идентификатором, создав его копию;

- Экспортировать — сохранить текущее правило в виде YAML-файла;
- Удалить — удалить открытое в редакторе правило.

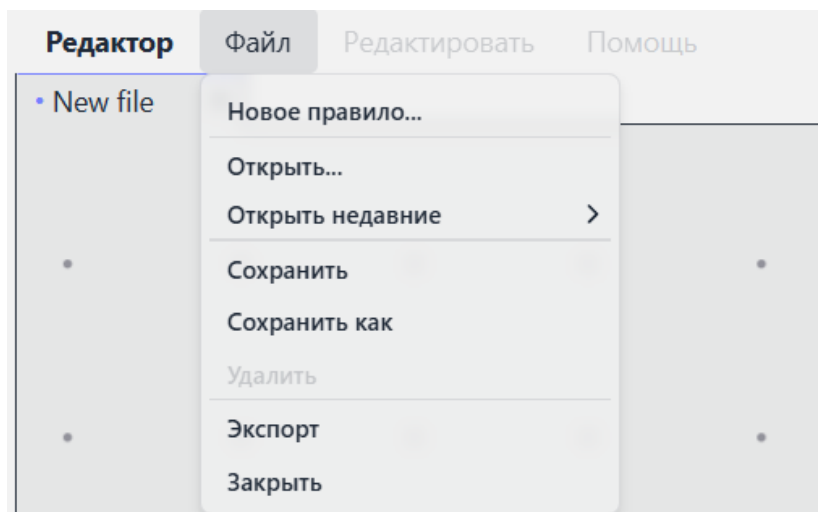


Рисунок 7 - Меню редактора правил

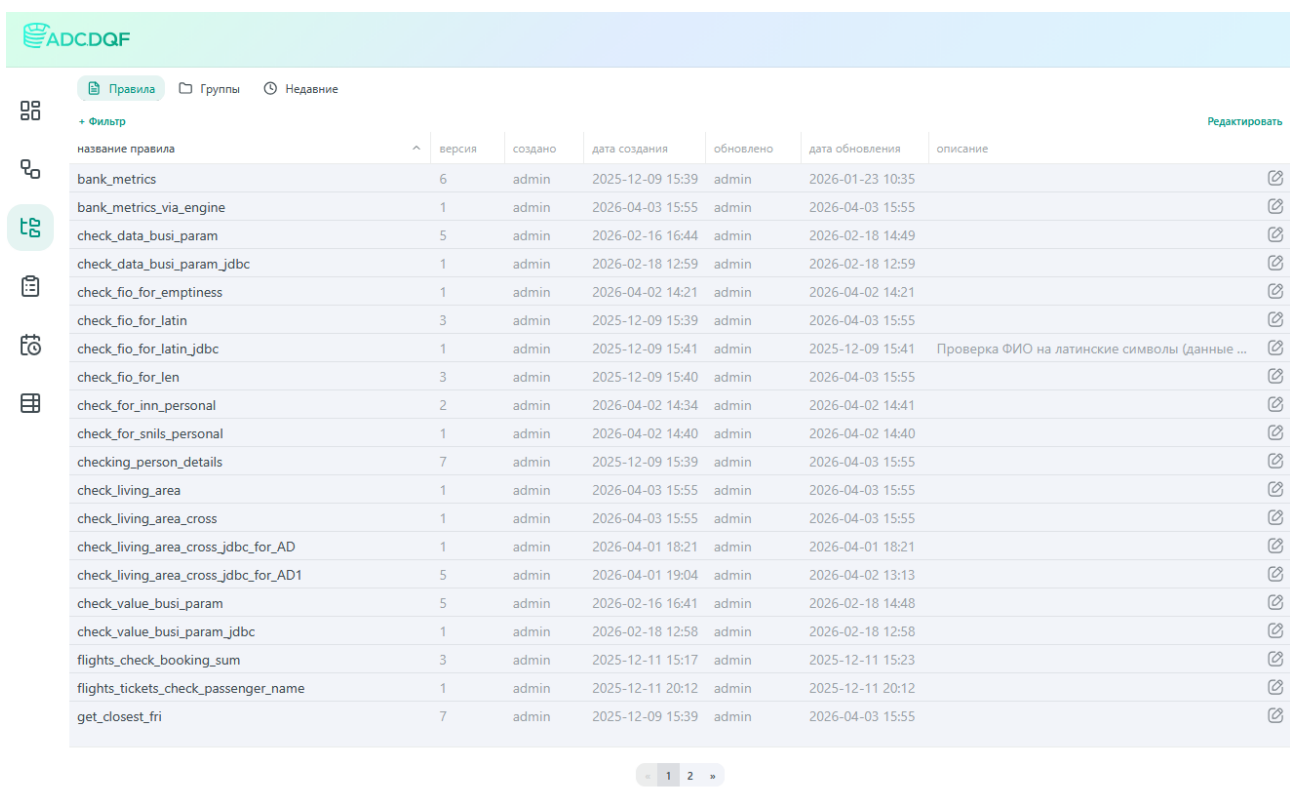
Все действия выполняются через графический интерфейс и доступны в верхней панели приложения.

Каталог правил

Каталога правил – это сервис, позволяющий пользователю управлять правилами и группами правил.

Список правил

В каталоге правил список представляется в виде таблицы с ключевыми параметрами каждого правила (Рисунок 8). Пользователи могут управлять этим списком, фильтруя и сортируя его по необходимым параметрам. Кроме того, с этой страницы можно открыть правило в редакторе, удалить выбранное правило или создать новое, с автоматическим переходом в редактор правил.



название правила	версия	создано	дата создания	обновлено	дата обновления	описание	Редактировать
bank_metrics	6	admin	2025-12-09 15:39	admin	2026-01-23 10:35		
bank_metrics_via_engine	1	admin	2026-04-03 15:55	admin	2026-04-03 15:55		
check_data_busi_param	5	admin	2026-02-16 16:44	admin	2026-02-18 14:49		
check_data_busi_param_jdbc	1	admin	2026-02-18 12:59	admin	2026-02-18 12:59		
check_fio_for_emptiness	1	admin	2026-04-02 14:21	admin	2026-04-02 14:21		
check_fio_for_latin	3	admin	2025-12-09 15:39	admin	2026-04-03 15:55		
check_fio_for_latin_jdbc	1	admin	2025-12-09 15:41	admin	2025-12-09 15:41	Проверка ФИО на латинские символы (данные ...	
check_fio_for_len	3	admin	2025-12-09 15:40	admin	2026-04-03 15:55		
check_for_inn_personal	2	admin	2026-04-02 14:34	admin	2026-04-02 14:41		
check_for_snils_personal	1	admin	2026-04-02 14:40	admin	2026-04-02 14:40		
checking_person_details	7	admin	2025-12-09 15:39	admin	2026-04-03 15:55		
check_living_area	1	admin	2026-04-03 15:55	admin	2026-04-03 15:55		
check_living_area_cross	1	admin	2026-04-03 15:55	admin	2026-04-03 15:55		
check_living_area_cross_jdbc_for_AD	1	admin	2026-04-01 18:21	admin	2026-04-01 18:21		
check_living_area_cross_jdbc_for_AD1	5	admin	2026-04-01 19:04	admin	2026-04-02 13:13		
check_value_busi_param	5	admin	2026-02-16 16:41	admin	2026-02-18 14:48		
check_value_busi_param_jdbc	1	admin	2026-02-18 12:58	admin	2026-02-18 12:58		
flights_check_booking_sum	3	admin	2025-12-11 15:17	admin	2025-12-11 15:23		
flights_tickets_check_passenger_name	1	admin	2025-12-11 20:12	admin	2025-12-11 20:12		
get_closest_fri	7	admin	2025-12-09 15:39	admin	2026-04-03 15:55		

Рисунок 8 - Каталог правил

Список групп

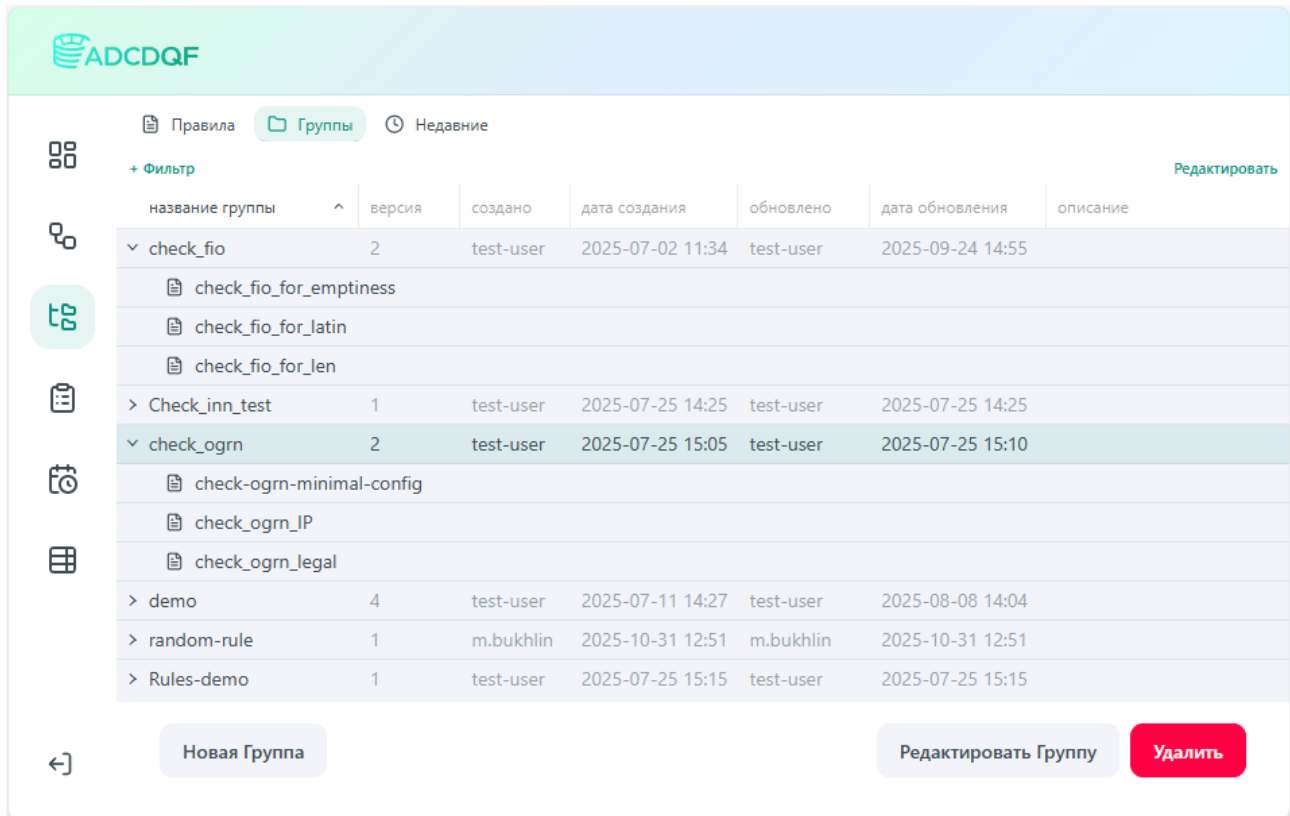
Группа правил – это совокупность правил, объединенных вместе в соответствии с бизнес-потребностью.

Например, проверка данных физического лица, как правило, требует не одного, а нескольких правил. В этом случае создаётся группа «Проверка физ. лиц», в которую могут входить правила, такие как «Проверка СНИЛС», «Проверка ФИО» и другие.

Группы упрощают организацию каталога и позволяют удобно управлять связанными наборами правил. Они могут использоваться при создании задачи на проверку данных, однако результаты по-прежнему формируются отдельно для каждого входящего в группу

правила. Таким образом, группа — это способ логически объединить правила, не влияя на их индивидуальную обработку и отчётность.

Список групп представлен в виде таблицы, аналогичной списку правил, и поддерживает сортировку и фильтрацию по ключевым параметрам (Рисунок 9). Это позволяет быстро находить нужные группы и управлять ими.



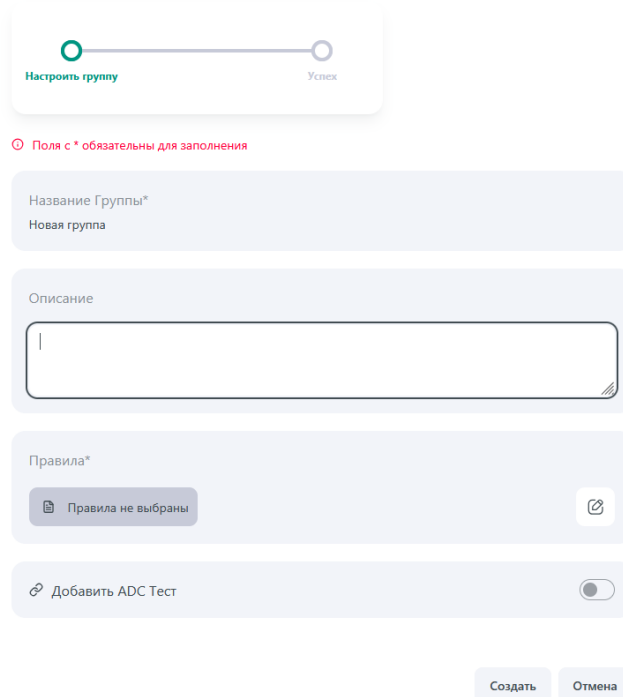
название группы	версия	создано	дата создания	обновлено	дата обновления	описание
check_fio	2	test-user	2025-07-02 11:34	test-user	2025-09-24 14:55	
check_fio_for_emptiness						
check_fio_for_latin						
check_fio_for_len						
Check_inn_test	1	test-user	2025-07-25 14:25	test-user	2025-07-25 14:25	
check_ogrн	2	test-user	2025-07-25 15:05	test-user	2025-07-25 15:10	
check_ogrн-minimal-config						
check_ogrн_IP						
check_ogrн_legal						
demo	4	test-user	2025-07-11 14:27	test-user	2025-08-08 14:04	
random-rule	1	m.bukhlin	2025-10-31 12:51	m.bukhlin	2025-10-31 12:51	
Rules-demo	1	test-user	2025-07-25 15:15	test-user	2025-07-25 15:15	

Рисунок 9 - Список групп в каталоге

Создание группы

Вы можете создать новую группу, указав список правил, которые должны в неё входить. Для этого необходимо выбрать правила из доступного каталога и присвоить группе уникальное имя (Рисунок 10).

Добавить группу правил



Настроить группу

Успех

Поля с * обязательны для заполнения

Название Группы*

Новая группа

Описание

Правила*

Правила не выбраны

Добавить ADC Тест

Создать Отмена

Рисунок 10 - Создание группы

Состав группы можно изменять в любое время: добавлять новые правила или удалять существующие.

Связь группы правил с тест-кейсами Arenadata Catalog

Группа правил может быть отправлена в модуль качества данных Arenadata Catalog (ADC), для этого необходимо указать следующую информацию:

- Владелец – владелец данного теста качества данных.
- Таблица – таблица, содержащая данные проверяемого объекта. К данному объекту будет привязан тест качества данных.
- Профили – тип теста. Можно выбрать несколько значений.
- Общее количество записей - максимальное количество записей, отправляемое из проверяемой таблицы на проверку. Обеспечивает возможность ограничить количество проверяемых строк таблицы. Рекомендуется указывать значение равное ожидаемому количеству строк проверяемой таблицы.
- Размер партии - количество записей, запрашиваемых из таблицы для отправки на проверку в одном запросе. Обеспечивает возможность управление нагрузкой на ADC.DQF и на источник данных. Рекомендуется указывать значение равное 500, либо рассчитывать исходя из технических характеристик планировщика задач.

После сохранения данная группа правил будет создана в ADC, после чего в ADC можно запустить проверки по данной группе правил и просматривать их агрегированные результаты.

Добавить ADC Тест [Тoggled On]

Владелец*

Владелец не назначен [Edit]

Таблица*

Таблица не выбрана

Профили*

Профили не выбраны

- Точность
- Полнота
- Соответствие
- Согласованность
- Целостность
- Разумность
- Своевременность
- Уникальность

Общее Количество Записей*

Размер Партии*

Создать Отмена

Рисунок 11 – Каталог правил. Форма создания группы, связанной с ADC

Планирование задач

Сервис планирования задач – это сервис, который создает задачи для проверки набора данных и формирует сообщения-запросы для Сервиса исполнения правил.

Задача — настраиваемый объект, предназначенный для выполнения группы правил на заданных данных.

В рамках одной задачи инициируется создание набора проверок: каждая проверка представляет собой применение одного правила к одному объекту данных. Таким образом, одна задача может включать множество проверок, соответствующих выбранной группе правил и входным данным.

Список задач

Все задачи представлены в виде таблицы, содержащей ключевые параметры каждой задачи (Рисунок 12). Пользователь может управлять списком, применяя фильтры и сортировку по нужным полям. Это позволяет быстро находить необходимые задачи и отслеживать их статус.

ADCDQF							
+ Фильтр							
	id	id группы	название группы	версия	дата создания	создано	статус
	1776248762100362	1776248762100045		1	2026-04-15 13:26	admin	в очереди
	1776211207100361	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-15 03:00	admin	отправлено
	1776169289100360	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-14 15:21	admin	отправлено
	1776124807100359	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-14 03:00	admin	отправлено
	1776038407100358	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-13 03:00	admin	отправлено
	1775952004100357	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-12 03:00	admin	отправлено
	1775865603100356	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-11 03:00	admin	отправлено
	1775779204100355	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-10 03:00	admin	отправлено
	1775692803100354	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-09 03:00	admin	отправлено
	1775606403100353	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-08 03:00	admin	отправлено
	1775569865100352	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-07 16:51	admin	отправлено
	1775566346100351	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-07 15:52	admin	отправлено
	1775563024100350	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-07 14:57	admin	отправлено
	1775550627100349	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-07 11:30	admin	отправлено
	1775520003100348	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-07 03:00	admin	отправлено
	1775506819100347	1775130482100041	Контроль корректности значения СНИЛС ФЛ	1	2026-04-06 23:20	admin	отправлено
	1775506819100346	1775130317100040	Контроль корректности значения ИНН ФЛ	1	2026-04-06 23:20	admin	отправлено
	1775506819100345	1775129389100039	Проверка ФИО физического лица.	1	2026-04-06 23:20	admin	отправлено
	1775506812100344	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-06 23:20	admin	отправлено
	1775506460100343	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-06 23:14	admin	отправлено

Рисунок 12 - Список задач

Создание задачи

Пользователь может создать новую задачу со страницы со списком задач. В процессе создания необходимо указать (Рисунок 13):

- данные, которые необходимо проверить;
- стратегию разбивки данных:

- AS IS — все переданные данные проверяются одним вызовом каждого правила в группе. Подходит для случаев, когда данные представляют собой единое значение и должны анализироваться как целое;
- SPLIT — каждый элемент входного массива обрабатывается отдельно: правило применяется к каждому элементу по отдельности. Используется, когда данные представляют собой список однотипных значений (например, перечень физических или юридических лиц), и требуется провести индивидуальную проверку для каждого из них.
- группу правил — можно выбрать существующую группу или задать набор правил, на основе которого система сформирует внутреннюю группу.

Создать задачу
✕

Стратегия ⊙

AS IS
SPLIT

Данные

```

1  [
2  {
3      "lastName": "Иванов",
4      "firstName": "Иван",
5      "patronymic": "Иванович"
6  }
7  {
8      "lastName": "Чжан",
9      "firstName": "Ю",
10     "patronymic": "Вей"
11 }
12 ]
```

Группа Правила

📄 Проверка ФИО физического лица.

📄
✕

Создать
Отмена

Рисунок 13 - Создание задачи

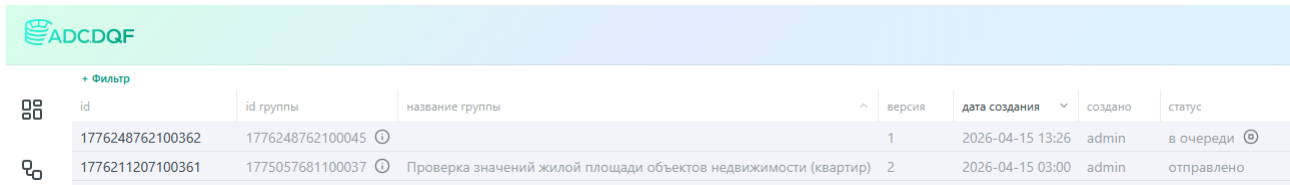
Изменение статуса задачи

Пользователь может отменить задачу, но только до момента, пока она не перешла в статус «Отправлено» (Рисунок 14).

Задача может находиться в одном из следующих статусов:

- Отправлено (SENT) — все сообщения задачи переданы во входящую очередь сервиса исполнения правил;

- В очереди (QUEUED) — сообщения задачи ещё не отправлены полностью или вовсе не отправлены;
- Отменено (CANCELLED) — задача отменена пользователем до начала отправки.

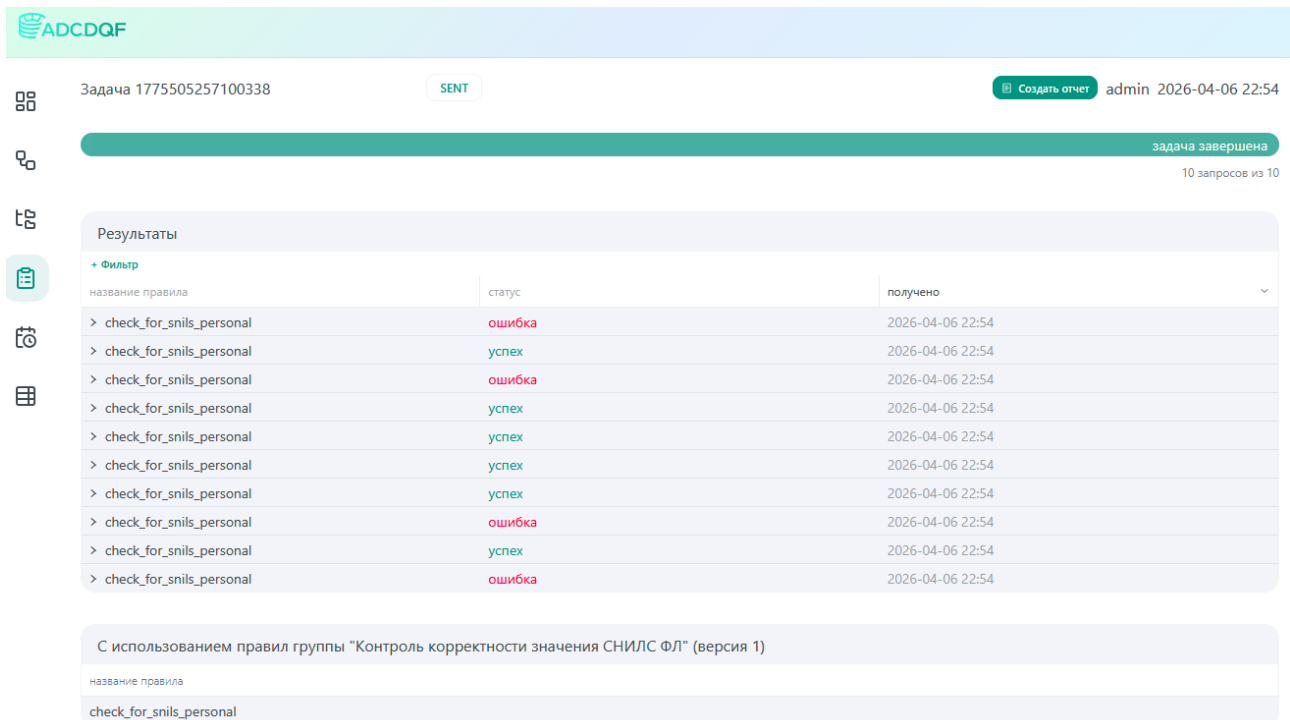


id	id группы	название группы	версия	дата создания	создано	статус
1776248762100362	1776248762100045		1	2026-04-15 13:26	admin	в очереди
1776211207100361	1775057681100037	Проверка значений жилой площади объектов недвижимости (квартир)	2	2026-04-15 03:00	admin	отправлено

Рисунок 14 - Остановка выполнения задачи

Просмотр результатов выполнения задачи

Со страницы списка задач можно открыть любую задачу и просмотреть результаты её выполнения (Рисунок 15).



Задача 177505257100338 SENT Создать отчет admin 2026-04-06 22:54

задача завершена
10 запросов из 10

название правила	статус	получено
> check_for_snils_personal	ошибка	2026-04-06 22:54
> check_for_snils_personal	успех	2026-04-06 22:54
> check_for_snils_personal	ошибка	2026-04-06 22:54
> check_for_snils_personal	успех	2026-04-06 22:54
> check_for_snils_personal	успех	2026-04-06 22:54
> check_for_snils_personal	успех	2026-04-06 22:54
> check_for_snils_personal	успех	2026-04-06 22:54
> check_for_snils_personal	ошибка	2026-04-06 22:54
> check_for_snils_personal	успех	2026-04-06 22:54
> check_for_snils_personal	ошибка	2026-04-06 22:54

С использованием правил группы "Контроль корректности значения СНИЛС ФЛ" (версия 1)

название правила
check_for_snils_personal

Рисунок 15 - Результаты выполнения задачи

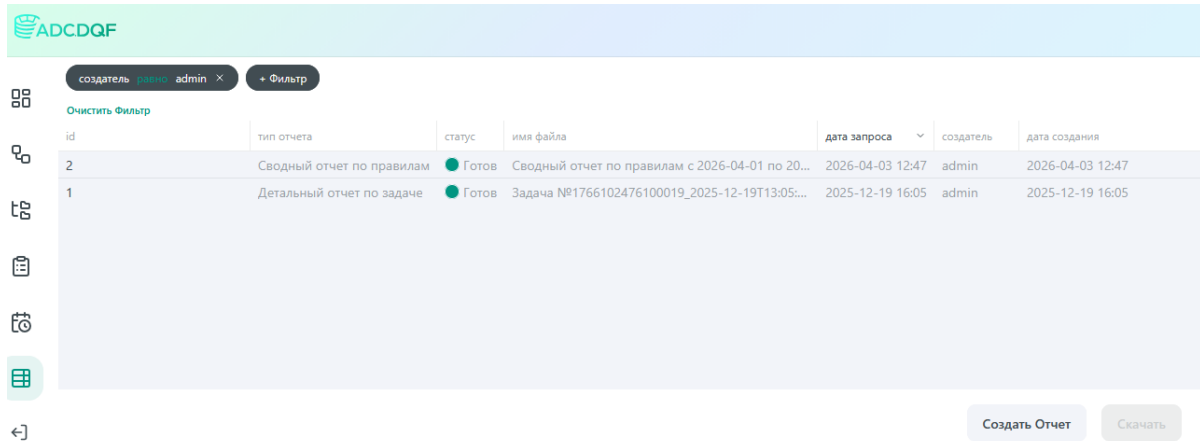
Результаты отображаются по каждому вызову правила отдельно. Для каждого вызова указаны:

- Статус выполнения:
 - Успех — ошибок в проверяемых данных не обнаружено;
 - Ошибка — выявлены ошибки в данных;
 - Прервано — при выполнении правила возникла техническая ошибка.
- Дополнительная информация, возвращаемая в результате выполнения правила. Она представлена в формате JSON и может содержать произвольные поля, заданные при конфигурации правила. Например, описание ошибки может быть возвращено в отдельном атрибуте:

- *"errorDescription": "Фамилия имя или отчество содержат латинские буквы"*

Отчетность

Для анализа результатов проверок пользователю доступны встроенные инструменты отчётности (Рисунок 16).

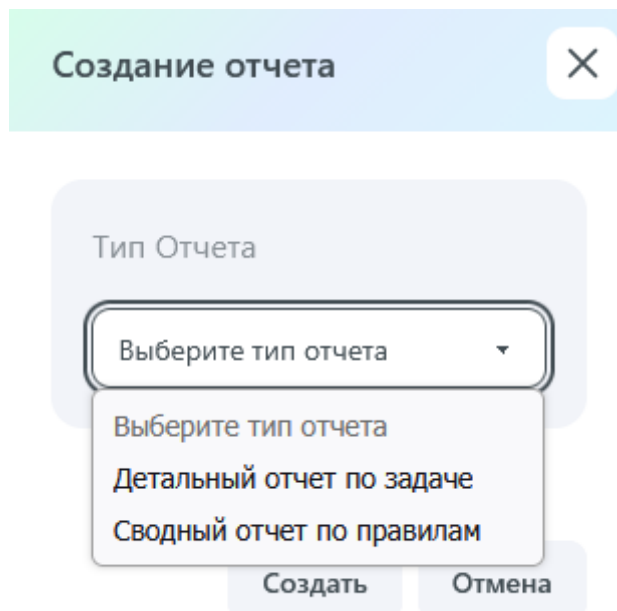


id	тип отчета	статус	имя файла	дата запроса	создатель	дата создания
2	Сводный отчет по правилам	Готов	Сводный отчет по правилам с 2026-04-01 по 20...	2026-04-03 12:47	admin	2026-04-03 12:47
1	Детальный отчет по задаче	Готов	Задача №1766102476100019_2025-12-19T13:05:...	2025-12-19 16:05	admin	2025-12-19 16:05

Рисунок 16 - Запросы на создание отчетов

Система позволяет сформировать один из следующих видов отчётов (Рисунок 17):

- Детальный отчёт по задаче — содержит полную информацию о каждом вызове правила внутри выбранной задачи. Этот отчёт практически полностью повторяет структуру страницы просмотра задачи и может использоваться для выгрузки или архивации данных.
- Сводный отчёт по правилам — включает статистику выполнения каждого правила за выбранный период времени: количество проверок со статусами «успех», «ошибка» и «прервано». Используется для общего анализа качества данных и выявления проблемных участков.



Создание отчета ✕

Тип Отчета

Выберите тип отчета ▾

Выберите тип отчета
 Детальный отчет по задаче

Выберите тип отчета
 Сводный отчет по правилам

Создать
Отмена

Рисунок 17 - Выбор типа отчета

Формирование и загрузка отчетов

Создание отчёта начинается с запроса на его формирование. Это можно сделать:

- со страницы со всеми запросами на отчёты — для обоих типов отчётов;
- напрямую со страницы задачи — для создания детального отчёта по конкретной задаче (Рисунок 18).

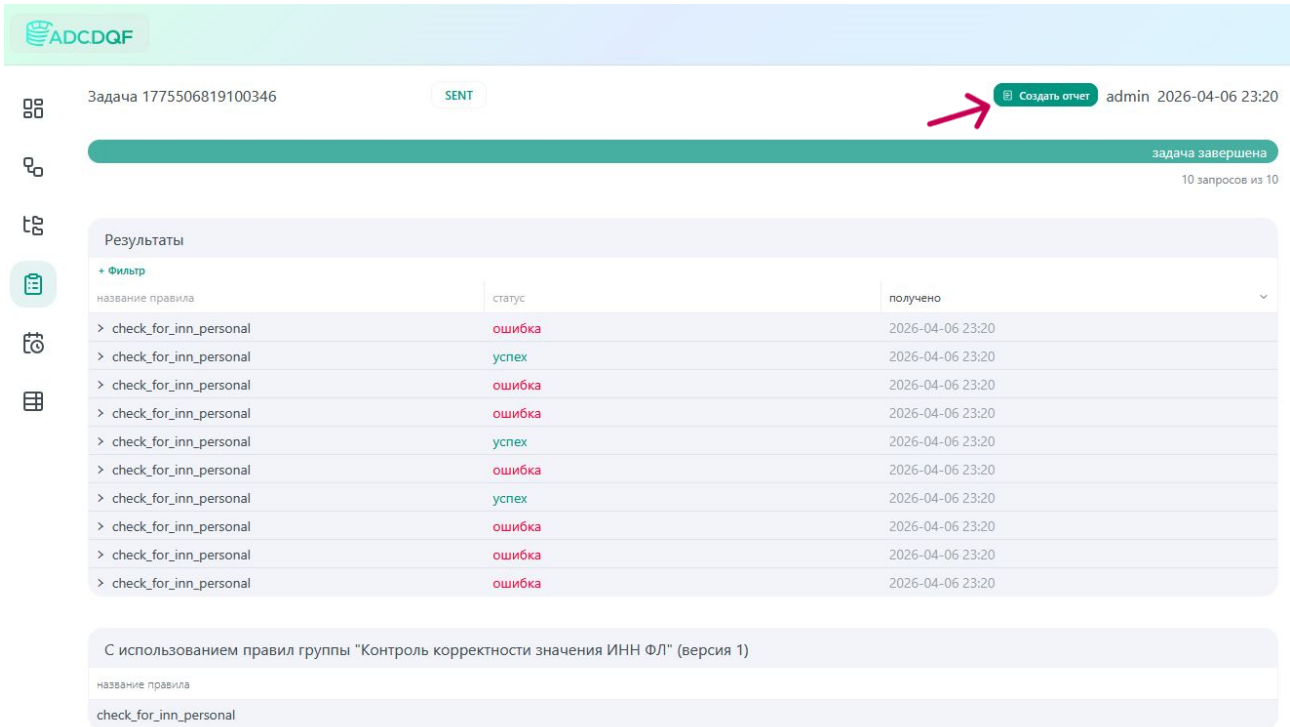


Рисунок 18 - Создание отчета со страницы задачи

После завершения формирования отчёт становится доступен для загрузки.

Все отчёты сохраняются в формате .xlsx и могут быть загружены повторно в любой момент со страницы с отчётами. Ниже представлен пример отчета типа «Детальный отчет по задаче» (Рисунок 19).

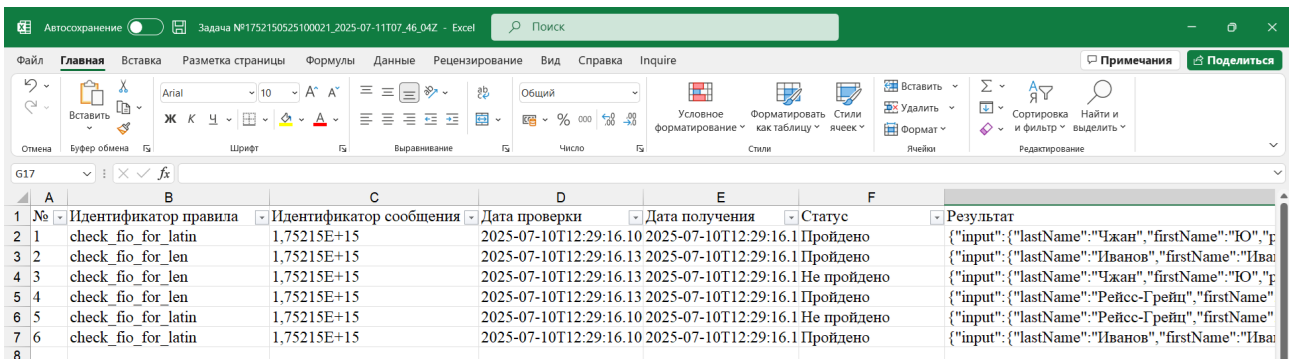


Рисунок 19 – Пример детального отчета по задаче

Дашборды

Для оценки результатов проверок и оперативного анализа качества данных в системе предусмотрена возможность создания дашбордов — интерактивных панелей с графиками и ключевыми метриками (Рисунок 20).

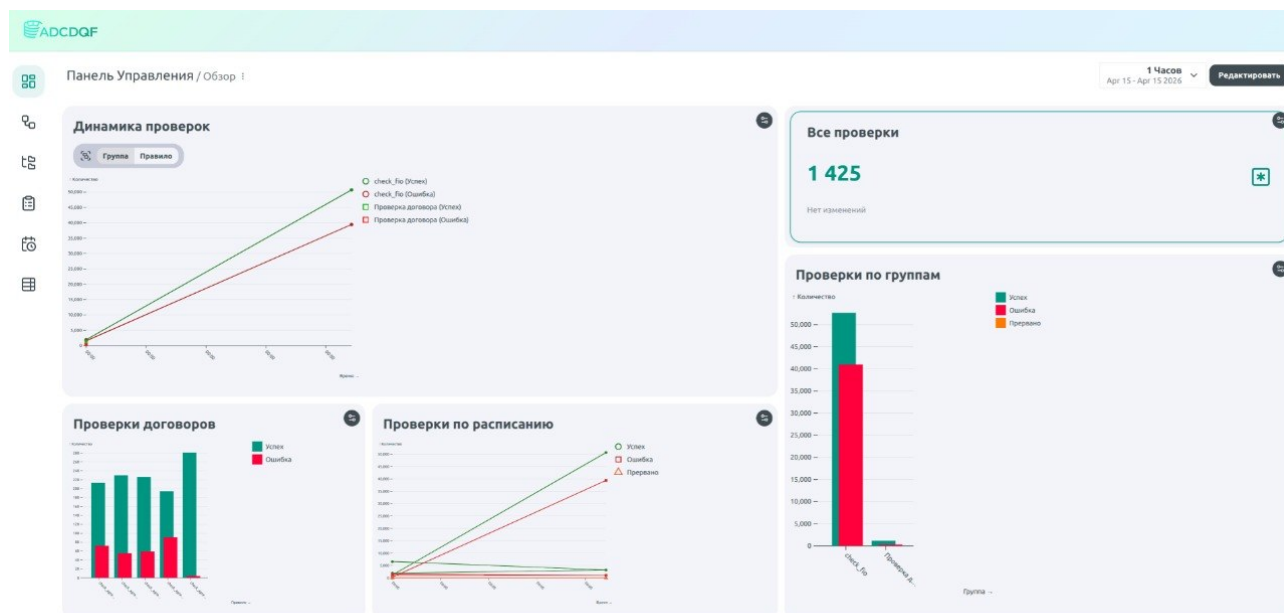


Рисунок 20 – Пример дашборда

Дашборды позволяют пользователям разных ролей — от технических специалистов до бизнес-аналитиков — отслеживать состояние и динамику проверок данных в удобном визуальном формате.

- Для аналитиков это — способ контролировать качество информации, выявлять проблемные участки и принимать решения на основе статистики.
- Для технических команд — инструмент наблюдения за стабильностью исполнения правил и возможностью быстро реагировать на возникающие ошибки.

Каждый пользователь может создать собственную панель с нужными графиками и показателями, адаптированной под его задачи. Гибкая настройка фильтров, уровней детализации и визуального представления делает дашборды мощным инструментом ежедневного контроля и анализа.

В приложении предусмотрена возможность создания и управления пользовательскими панелями дашбордов.

Пользователь может (Рисунок 21):

- создать новую панель управления;
- открыть из существующих;
- удалить любую из доступных панелей.

Открыть					
ID	Имя	Дата создания	Создано	Дата обновления	Обновлено
1775122485100002	ADC Demo Правила и группы	2026-04-02 12:34	admin	2026-04-02 12:38	admin

[Создать Панель Управления](#)
[Открыть](#)

Рисунок 21 - Управление дашбордами

Добавление и изменение виджетов

На панель можно добавить один из трёх типов виджетов:

- Линейная диаграмма — отображает изменение показателей во времени (Рисунок 22);
- Столбчатая диаграмма — показывает распределение проверок по выбранному параметру (Рисунок 23);
- Карточка — компактный индикатор с численным значением.

Настройки графиков (линейного и столбчатого):

- Имя виджета — произвольное название, задаваемое пользователем;
- Параметр по оси X — для *линейного* графика доступна только дата/время, для *столбчатого* — распределение по правилам, группам или расписаниям;
- Параметр по оси Y — возможен выбор одного или нескольких значений: количество успехов, ошибок и прерванных проверок;
- Иерархия — настройка drill-down: можно указать начальный уровень (расписания, группы, правила);
- Фильтр — по нужным правилам, группам или расписаниям.



Рисунок 22 – Линейная диаграмма

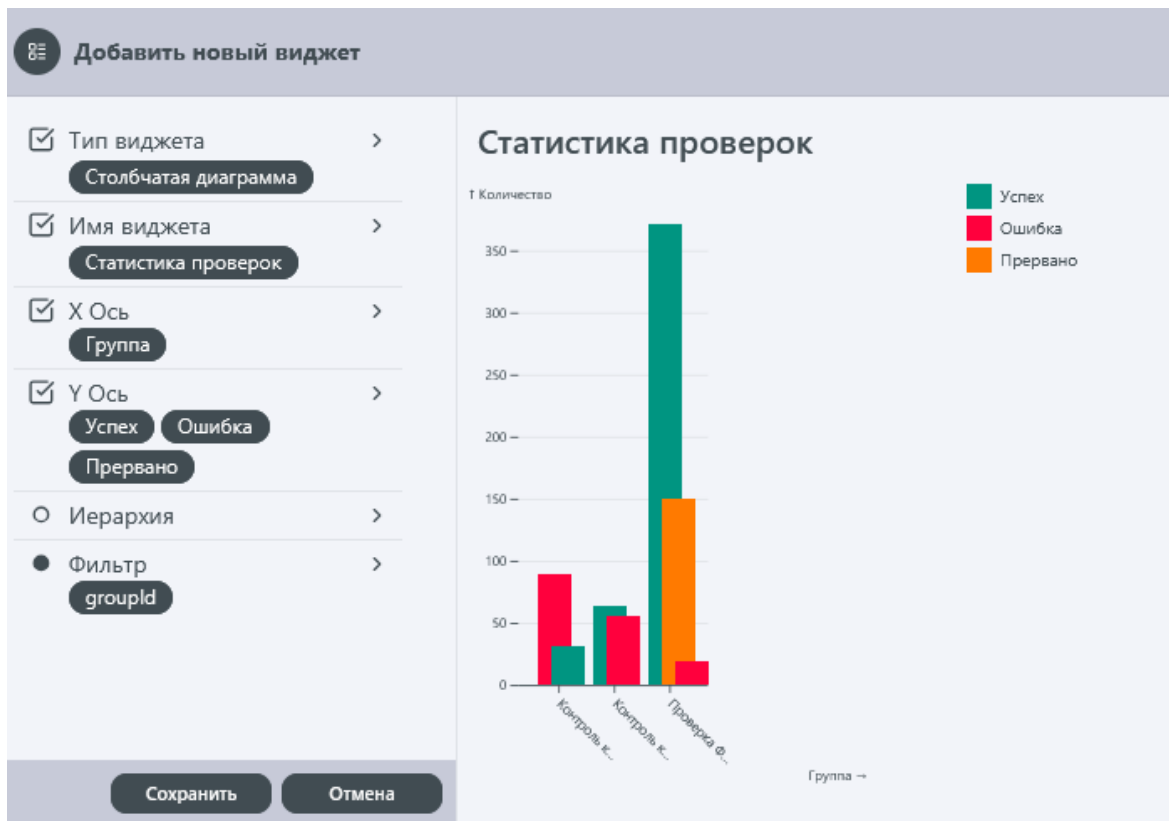


Рисунок 23 – Столбчатая диаграмма

Настройки карточки (Рисунок 24):

- Имя виджета;
- Показатель — один из трёх: количество успехов, ошибок или прерванных проверок;
- Фильтр — по выбранным правилам, группам или расписаниям.

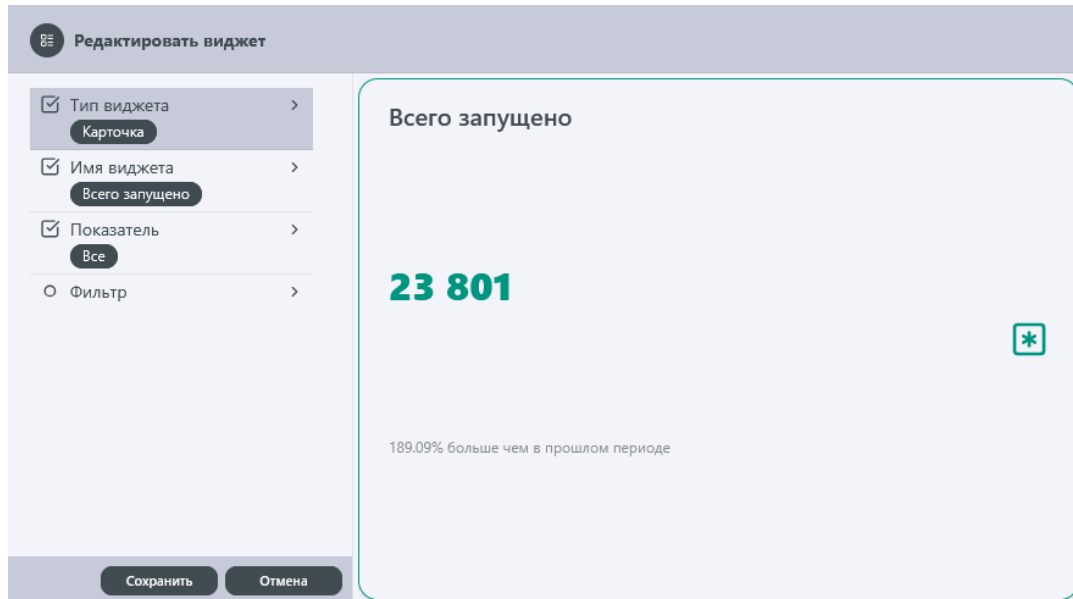


Рисунок 24 - Карточка

Редактирование дашборда

Редактирование панели включает:

- изменение расположения виджетов (drag-and-drop) (Рисунок 25),
- задание общего временного интервала, на основании которого строятся все графики панели,
- изменение имени панели.

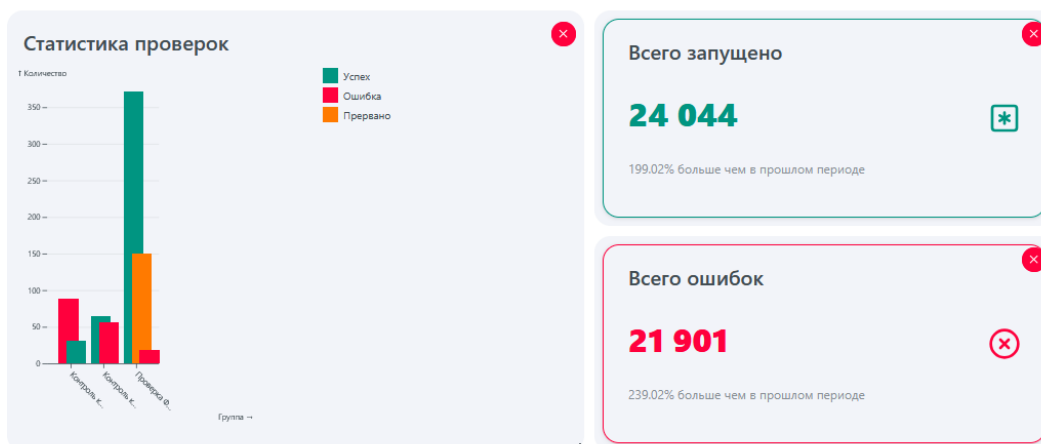


Рисунок 25 - Перемещение виджета на дашборде

Сценарии взаимодействия с приложением

Платформа предоставляет широкий набор возможностей для работы с качеством данных, охватывая все этапы — от проектирования проверок до анализа их результатов. В зависимости от роли пользователя в организации, сценарии взаимодействия с приложением могут существенно различаться.

В этом разделе представлены типовые кейсы использования системы с учётом ролей участников процесса. Они помогут лучше понять, как отдельные компоненты платформы могут использоваться для решения практических задач — от обеспечения корректности данных до управления стратегией качества на уровне всей компании.

Владелец процесса качества данных

Задачи: отвечает за формирование и контроль выполнения политики качества данных в организации. Участвует в согласовании правил и метрик, принимает управленческие решения на основе данных.

Сценарий 1: «Мониторинг реализации политики качества данных».

1. Заходит в дашборд, чтобы оценить динамику числа ошибок по ключевым группам проверок за месяц;
2. Делает drill-down до конкретных правил, где наблюдается рост количества несоответствий;
3. Формирует сводный отчет по этим правилам за последние 3 месяца для анализа в Excel;
4. На основе данных и трендов инициирует изменение набора правил или уточнение логики.

Аналитик качества данных

Задачи: проектирует и реализует правила проверки данных, отвечает за их корректность и актуальность, участвует в анализе результатов проверок и их интерпретации.

Сценарий 1: «Разработка и тестирование нового правила проверки».

1. Создает новое правило в редакторе, используя базовые алгоритмы и механизмы управления потоком;
2. Добавляет вложенное правило и параметризует все блоки правила;
3. Вводит пример входных данных и запускает проверку правила;
4. Проверяет выходной результат, выявляет ошибку в логике, исправляет выражение в одном из базовых алгоритмов;
5. Сохраняет правило в каталог, добавляет его в соответствующую группу.

Сценарий 2: «Актуализация существующих проверок».

1. Открывает правило из каталога, проверяет его соответствие обновленной бизнес-логике;
2. Вносит изменения в правило;
3. Сохраняет новую версию правила;
4. Использует тестирование правила, чтобы убедиться в корректности результата.

Владелец источника данных

Задачи: отвечает за предоставление и корректность данных, передаваемых в систему. Его цель — своевременно выявлять и устранять несоответствия на своей стороне.

Сценарий 1: «Контроль за качеством перед выгрузкой данных».

1. Создает задачу проверки данных, загружая массив объектов из своей системы;
2. Выбирает нужную группу правил;
3. Отслеживает статус задачи, затем просматривает детализированные результаты выполнения;
4. Находит объекты, не прошедшие проверку, выгружает json с доп. информацией об ошибках;
5. Вносит изменения в источник данных перед публикацией информации.

Потребитель данных

Задачи: использует данные в своей работе (например, для отчетности, аналитики или операционных решений). Заинтересован в получении проверенной и корректной информации.

Сценарий 1: «Проверка данных перед использованием в отчете».

1. Пользователь открывает ранее созданный дашборд, содержащий информацию о качестве данных;
2. Добавляет на панель виджет в формате столбчатой диаграммы, отображающий количество непройденных проверок с разбиением по группам правил;
3. Анализирует результаты и убеждается, что по интересующим его группам данных за последний период не зафиксировано критичных нарушений;
4. Использует проверенные и надежные данные для подготовки внешнего BI-отчета или включения в аналитическую презентацию.