

Руководство пользователя и  
функциональные характеристики MegaRITM

## Оглавление

<b>1</b>	<b>Введение</b>	<b>4</b>
1.1	Описание системы	4
1.2	Параметры подключения	4
1.3	Глоссарий	4
1.4	Логическая модель	7
<b>2</b>	<b>Объекты системы</b>	<b>8</b>
2.1	Главная страница	8
2.2	Меню навигации	8
2.3	Концепция конфигурации	8
2.4	Канал	9
2.5	Точка взаимодействия	10
2.6	Группа сообщений	12
2.7	Сообщение	14
2.8	Атрибуты персональной настройки	15
2.9	Категории атрибутов персональной настройки	16
2.10	Категории атрибутов принятия решения	17
2.11	Атрибуты принятия решения	17
2.12	Правила	18
2.13	Область	20
2.14	Арбитраж	21
2.15	Цель	21
2.16	Стратегия	23
<b>3</b>	<b>Работа с компонентами</b>	<b>26</b>
3.1	Бизнес среда	26
3.2	Продвижение компонентов	26
3.3	Удаление компонентов	28
3.4	Восстановление компонентов	29
<b>4</b>	<b>Работа с правилами</b>	<b>31</b>
4.1	Логика правил	31
4.2	Компоненты, которые могут иметь правила	31
4.3	Конструктор правил	32
4.4	Критерии отбора и условия	32
4.5	Настройка правил	33
4.5.1	Правило с несколькими атрибутами и объединением атрибутов в блоки	33
4.5.2	Правило с датами	34
4.5.3	Правило с выражением	35
<b>5</b>	<b>Настройка деревьев</b>	<b>37</b>
5.1	Переход к графическому режиму	37
5.2	Работа в редакторе дерева	37
5.3	Добавление пунктов назначения	38
5.4	Удаление узлов	38
5.5	Редактирование дерева	38
5.5.1	Добавление правил	39
5.5.2	Логика управления потоком	39
5.5.3	Изменение порядка веток	39
5.5.4	Изменение источника	40
5.6	Сохранение изменений	40
5.7	Дополнительные настройки	41
<b>6</b>	<b>Механизм принятия решения</b>	<b>43</b>
6.1	Проверка правил	43
6.2	Арбитраж	43
6.3	Стратегии	43
6.4	Ограничение количества сообщений	44
<b>7</b>	<b>Сервис отсечений</b>	<b>45</b>
7.1	Назначение	45
7.2	Открытие сервиса отсечений	45

7.3	Работа с фильтром.....	46
<b>8</b>	<b>Администрирование.....</b>	<b>47</b>
8.1	Добавление пользователей.....	47
8.2	Просмотр авторизованных пользователей.....	47
8.3	Импорт данных.....	47
8.4	Импорт (RITM).....	49
8.5	Экспорт (RITM).....	49
<b>9</b>	<b>Функции выражения.....</b>	<b>51</b>

# 1 Введение

Документ предназначен для обеспечения правильного и эффективного использования информационной системы MegaFon Real-time Interaction Tree Manager (MegaRITM, далее по тексту Система).

В нем представлено описание системы, её функции и порядок работы. Руководство разработано для конечных пользователей и служит справочным материалом при работе с Системой.

## 1.1 Описание системы

MegaRITM - это система по управлению маркетинговыми кампаниями, осуществляющая выбор маркетинговых предложений в режиме реального времени.

MegaRITM можно рассматривать как сервис принятия решений. Когда абонент взаимодействует с каналом продаж, канал запрашивает предложения, MegaRITM должен решить, какое предложение отправить для конкретного абонента в данном случае.

Система поддерживает следующие варианты работы с точки зрения предоставления абонентам предложений в реальном времени:

- Синхронный режим.

Абонент самостоятельно инициирует коммуникацию, используя доступный ему канал коммуникации. Например, Личный кабинет или входящие обращения в call center. В этом случае Система предоставляет абоненту предложения через тот же канал коммуникации и в той же сессии (например, сессия работы в личном кабинете или разговор с оператором call center). В данном режиме допускается предоставление абоненту нескольких предложений на один запрос.

- Асинхронный режим.

Абонент выполняет какое-либо действие (не обращается в канал коммуникации), в ответ на которое Система формирует и направляет через канал коммуникации предложение. В данном режиме допускается предоставление абоненту только одного предложения на один запрос. Пример асинхронной коммуникации – баланс абонента переходит через некоторый порог, Система подбирает и направляет абоненту через SMS предложение воспользоваться услугой Обещанный платеж.

## 1.2 Параметры подключения

Подключение к Системе осуществляется с помощью браузера с технологического рабочего стола UTAG (Unified Technical Access Gateway). Адрес для подключения <https://megaritm.megafon.ru/>

Для авторизации в системе используются доменные учетные записи.

## 1.3 Глоссарий

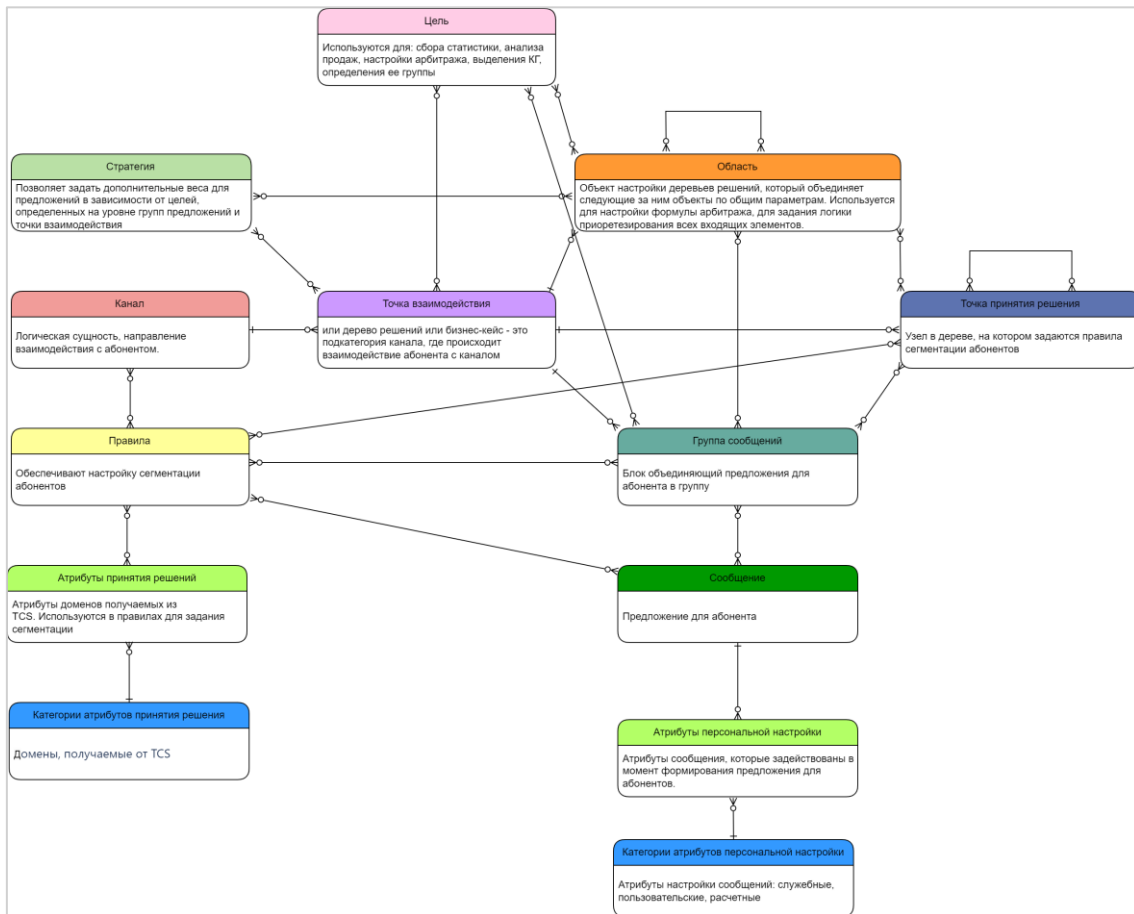
Данный документ использует следующие термины и определения:

Термин	Определение
Канал	Объект, объединяющий несколько деревьев решений, предназначенных для общего способа взаимодействия с абонентом.

Термин	Определение
Дерево решений (Точка взаимодействия)	Объект, содержащий взаимосвязи между правилами и сообщениями для построения логики выбора самого релевантного предложения клиенту.
Точка принятия решения	Узел дерева, позволяющий маршрутизировать абонента в соответствии с пройденными правилами.
Область	Узел дерева, позволяющий настроить список сообщений, доступных для возврата абоненту в конкретной ветке.
Группа сообщений	Узел дерева, позволяющий настроить список содержащий в себе сообщения, которые могут быть возвращены абоненту, если он удовлетворяет всем правилам, ведущим к данной группе сообщений.
Сообщение (Предложение)	Предложение абоненту с настроенными атрибутами персональной настройки, входящими в данное предложение.
Тип предложения	Атрибут предложения, определяющие принадлежность к продуктовой группе (Core, VAS).
Правило	Логическое выражение, позволяющее провести сегментацию абонента по определенному признаку.
Арбитраж	Процесс приоритизации отобранных сообщений.
Стратегия	Объект, связывающий правило, балансировщик целей и производящий отбор 1 или нескольких самых релевантных предложений.
Цель (Метка)	Метка, добавляемая на точку взаимодействия, область, группу сообщений. Используется для арбитража, тестирования и аналитики работы дерева.
Балансировщик целей	Механизм изменения веса цели, при расчете арбитража.
Атрибут принятия решений	Некоторая характеристика абонента, на основе значения которой может приниматься решение о выборе предложения для абонента.
Атрибут персональной настройки	Некоторая характеристика предложения, которая может отличаться и замещается реальным значением в итоговом предложении абоненту.
Категория предложения	Атрибут предложения, позволяющий сгруппировать предложения для выделения контрольных групп.
История контактов	Домен, содержащий исторические данные о направляемых абонентам предложениях.
Синхронная коммуникация	Коммуникация с абонентами в рамках входящих (pull) каналов коммуникаций. Предложения направляются абоненту в момент его обращения через тот же канал, в который обращается абонент.
Асинхронная коммуникация	Коммуникация с абонентами в рамках исходящих (push) каналов коммуникаций. Предложения направляются абоненту в момент определенного действия абонента.
Продуктивная эксплуатация	Эксплуатация Системы конечными пользователями на реальных данных Компании в реальном режиме работы.

Термин	Определение
MegaRITM	Система по управлению маркетинговыми кампаниями, осуществляющая выбор маркетинговых предложений в реальном времени.
Промоут	Перемещение объекта и связанных характеристик между разными средами (DEV, QA, PROD). То же, что и "Продвижение".
Продвижение	Перемещение объекта и связанных характеристик между разными средами (DEV, QA, PROD). То же, что и "Промоут".
CIM (Customer Interaction Manager)	Система управления маркетинговыми кампаниями в пакетном режиме.
RTM	Система подбора персональных маркетинговых предложений, заменяемая данным решением.
СУМК	Комплекс систем по управлению маркетинговыми кампаниями.
Единый канал, UC	Платформа Luxms MMS Sender, единая система для взаимодействия с каналами коммуникации.
Контрольная группа (КГ)	Группа абонентов, выбранная системой, для которой система не отправляет маркетинговые коммуникации и только фиксирует историю контактов. Используется для анализа запусков и корректировки результатов целевой группы.
Целевая группа (ЦГ)	Группа абонентов, выбранная системой, для которой система отправляет маркетинговые коммуникации и сохраняет историю отклика.

## 1.4 Логическая модель



## 2 Объекты системы

### 2.1 Главная страница

При входе в Систему, после успешной авторизации, открывается главная страница. Здесь отображен подсчет количества основных объектов системы. Нажав на любой из блоков, можно перейти к списку соответствующих объектов и их конфигурированию.

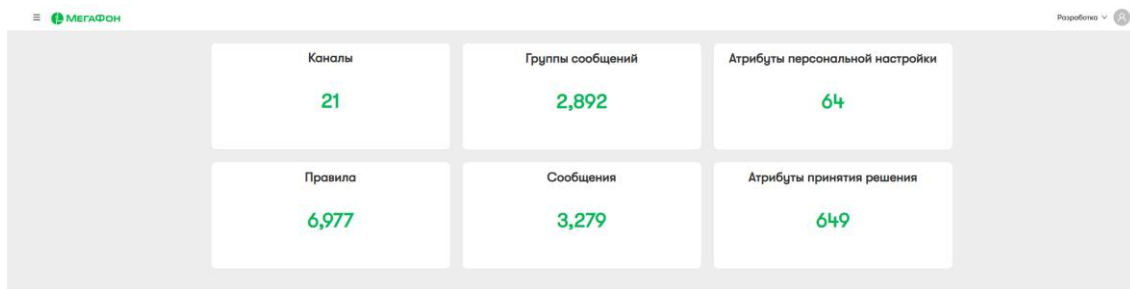


Рисунок 1.1. Главная страница

### 2.2 Меню навигации

Вы можете развернуть панель навигации, нажав на значок возле эмблемы МегаФон в верхнем левом углу. Чтобы свернуть панель навигации, нажмите значок ещё раз.

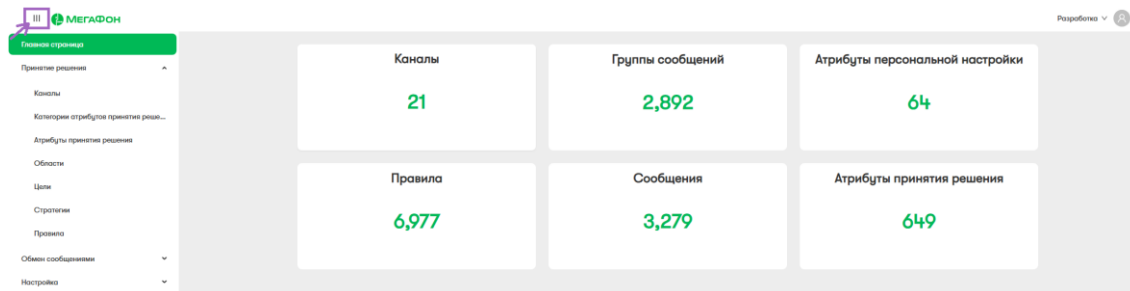


Рисунок 1.2. Меню навигации

Чтобы перейти к списку объектов из меню навигации нажмите на строку с необходимым типом объектов.

### 2.3 Концепция конфигурации

В следующих разделах приведены определения каждого из объектов и поясняется, как их настроить.

Несмотря на то, что существует множество объектов, которые могут быть задействованы, минимальная конфигурация, необходимая для отправки сообщений клиенту, следующая:

- Канал
- Точка взаимодействия
- Группа сообщений
- Сообщение

Каналы и Точки взаимодействия являются высшими элементами иерархии конфигурации. В одном Канале может быть 0 - N Точек взаимодействия. Точка взаимодействия может принадлежать только одному Каналу.

Сообщение должно входить в Группу сообщений, которая привязывается к Точке взаимодействия.

Группы сообщений и сами Сообщения могут быть связаны с несколькими Точками взаимодействия в Системе. При изменении конфигурации объекта, он изменится везде, где используется.

Полный список объектов, который можно использовать:

- Канал
- Точка взаимодействия
- Точка принятия решения (существует только внутри иерархии дерева)
- Область
- Группа сообщений
- Сообщение
- Правило
- Категории атрибутов принятия решения
- Атрибуты принятия решения
- Стратегия
- Цель
- Категории атрибутов персональной настройки
- Атрибуты персональной настройки

## 2.4 Канал

Объект, объединяющий несколько Точек взаимодействия, предназначенных для общего способа взаимодействия с абонентом.

Канал находится на вершине иерархии конфигурации.

Примеры Каналов, работающих в синхронном режиме: Личный кабинет, USSD (\*100#), Интернет магазин.

Все асинхронные взаимодействия выполняются в отдельно выделенном канале.

Наименование	Описание	Имя	Дата изменения
IVR		ivrbotall	29.06.2021 17:04:26
IVR Елена		ekaterina.gubenko	07.11.2024 12:58:06
СМС-Б		anna.akhmetova	27.06.2023 07:50:37
USSD *100	USSD *100#	ussd100	14.11.2023 14:58:27
Игры в МЛК Подарки		vladislav.vodunov	20.08.2022 00:01:21

Рисунок 1.3. Список Каналов

На форме со списком доступны фильтрация и сортировка по полям. Фильтрация реализована с поддержкой регулярных выражений в стандарте PCRE2, сравнение с выражением происходит нечувствительно к регистру. При выделении строки появляются кнопки возможных действий с объектом. Доступен множественный выбор строк. Для открытия контекста конкретного Канала необходимо дважды кликнуть на строку с объектом.

Для создания нового Канала необходимо нажать на кнопку добавления записи. Откроется модальная форма с полями:

- "Наименование" является обязательным для заполнения. Должно быть уникальным среди объектов с одним типом.
- "Наименование ключа" является обязательным для заполнения, заполняется автоматически на основе значения в поле "Наименование", доступно для редактирования в моменте создания, после сохранения объекта значение в поле "Наименование ключа" недоступно для редактирования. Должно быть уникальным среди объектов с одним типом.
- "Описание" не является обязательным для заполнения.

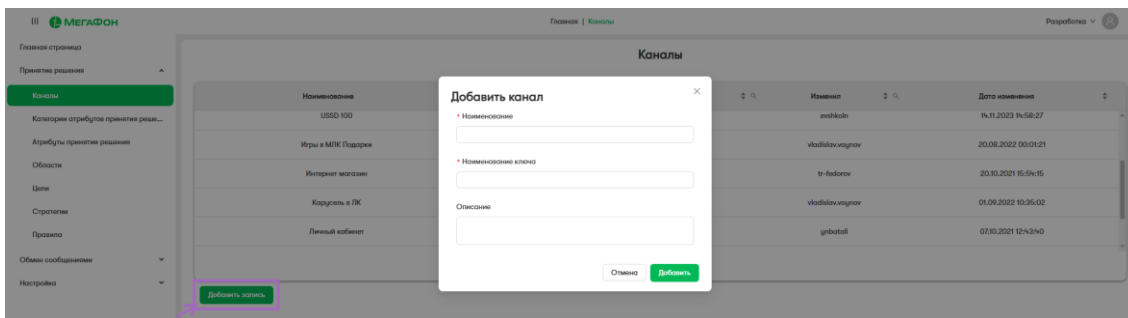


Рисунок 1.4. Создание Канала

После добавления записи открывается детальная форма объекта.

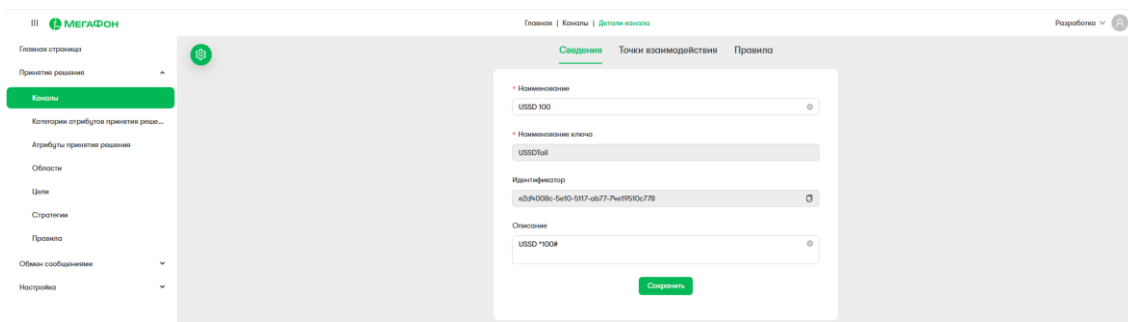


Рисунок 1.5. Канал

В контексте Канала представлены табы:

- "Сведения" отображает основную информация о Канале.
- "Точки взаимодействия" отображает список связанных с данным каналом Точек взаимодействия.
- "Правила" список Правил заданных на Канал.

## 2.5 Точка взаимодействия

Это подкатегория канала, в ней происходит взаимодействие абонента с каналом. Точку взаимодействия можно назвать деревом решений или бизнес-кейсом. Отдельного списка всех Точек взаимодействия нет, переход к конкретному объекту всегда осуществляется через Канал, которому он принадлежит.

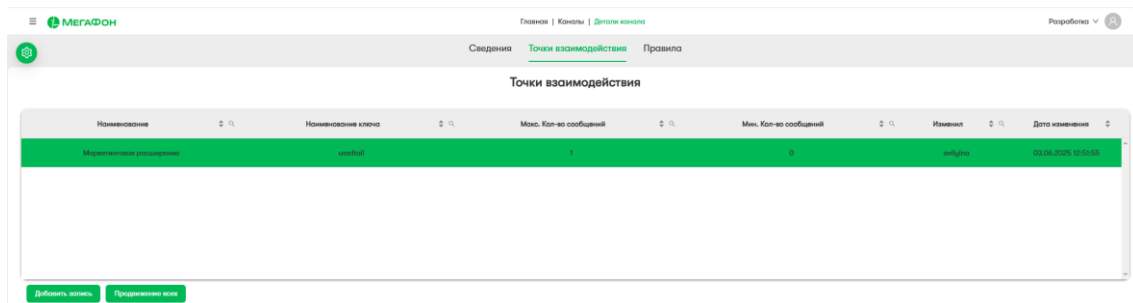


Рисунок 1.6. Список Точек взаимодействия в Канале

На форме со списком доступны фильтрация и сортировка по полям. Фильтрация реализована с поддержкой регулярных выражений в стандарте PCRE2, сравнение с выражением происходит нечувствительно к регистру.

При выделении строки появляются кнопки возможных действий с объектом. Доступен множественный выбор строк. Для открытия контекста конкретной Точки взаимодействия необходимо дважды кликнуть на строку с объектом.

Для создания новой Точки взаимодействия необходимо нажать на кнопку добавления записи. Откроется модальная форма с полями:

- "Наименование" является обязательным для заполнения. Должно быть уникальным среди объектов с одним типом.
- "Наименование ключа" является обязательным для заполнения, заполняется автоматически на основе значения в поле "Наименование", доступно для редактирования в моменте создания, после сохранения объекта значение в поле "Наименование ключа" недоступно для редактирования. Должно быть уникальным среди объектов с одним типом.
- "Описание" не является обязательным для заполнения.

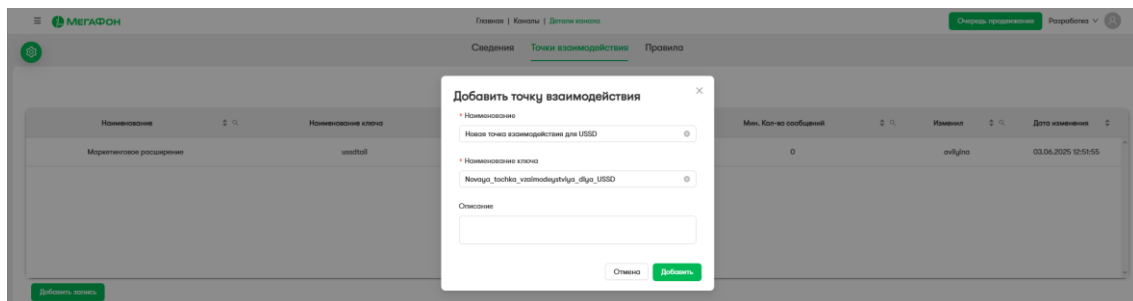


Рисунок 1.7. Создание Точки взаимодействия

После добавления записи открывается детальная форма объекта.

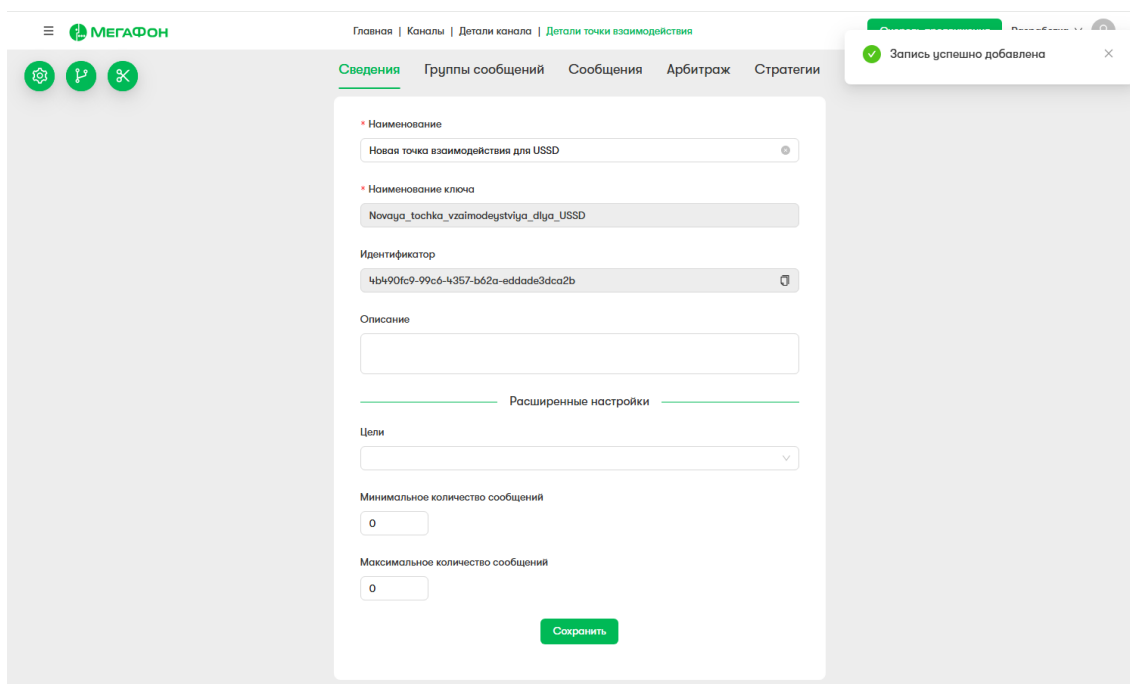


Рисунок 1.8. Точка взаимодействия

В контексте Точки взаимодействия представлены табы:

- "Сведения" отображает основную информация о Точке взаимодействия. После создания объекта можно навешать Цели для Точки взаимодействия, задать ограничения на минимальное и максимальное количество Сообщений отдаваемых абоненту.
- "Группы сообщений" список связанных с данной Точкой взаимодействия Групп сообщений.
- "Сообщения" список всех Сообщений относящихся к этой Точке взаимодействия.
- "Арбитраж" позволяет задать приоритезацию подобранных Сообщений на уровне Точки взаимодействия. Не является самостоятельным объектом.
- "Стратегии" позволяют корректировать оценку сообщения (ранее рассчитанную арбитражем) на основе целей.
- "История изменений" отображает историю продвижения дерева и включенных в него объектов (Правил, Групп сообщений, Сообщений, Областей, Стратегий).

## 2.6 Группа сообщений

Блок объединяющий предложения для абонента в группу.

Примерами Групп сообщений могут быть «Предложения к праздникам» или «Скидки к началу учебного года».

Группы сообщений находятся ниже точек взаимодействия в иерархии конфигурации, поэтому вы связываете группы сообщений с Точками взаимодействия. Связывание выполняется при редактировании Точки взаимодействия в графическом режиме, который будет описан позже.

Для создания новой Групп сообщений необходимо перейти к общему списку всех Групп сообщений и нажать на кнопку добавления записи. Откроется модальная форма с полями:

- "Наименование" является обязательным для заполнения. Должно быть уникальным среди объектов с одним типом.

- "Наименование ключа" является обязательным для заполнения, заполняется автоматически на основе значения в поле "Наименование", доступно для редактирования в момент создания, после сохранения объекта значение в поле "Наименование ключа" недоступно для редактирования. Должно быть уникальным среди объектов с одним типом.
- "Описание" не является обязательным для заполнения.
- "Владелец"

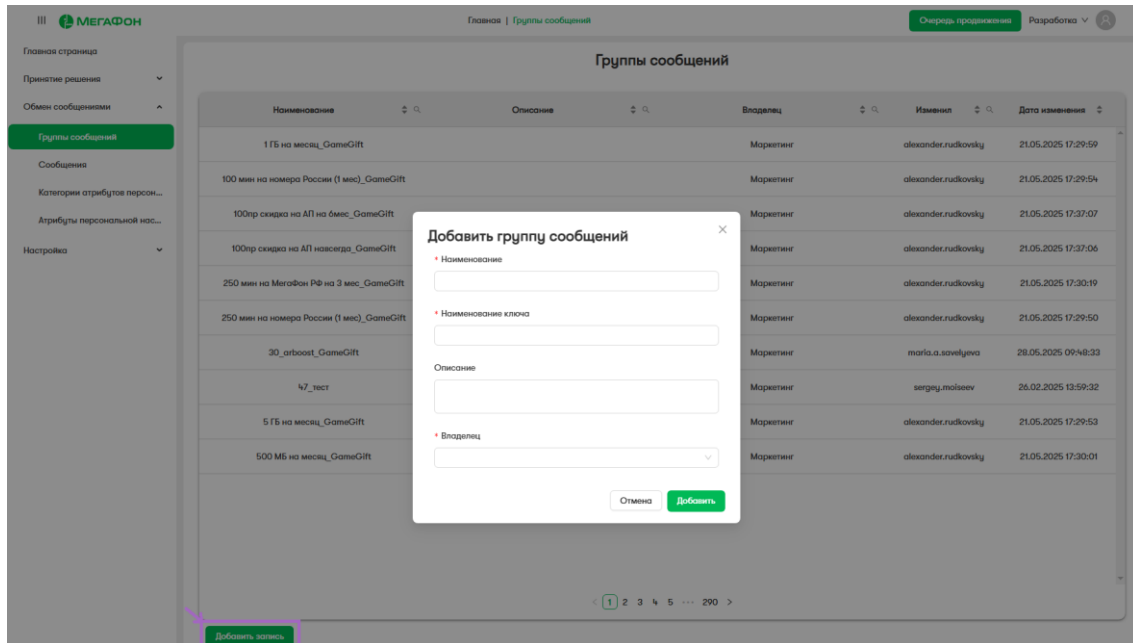


Рисунок 1.9. Создание Группы сообщений

После добавления записи открывается детальная форма объекта.

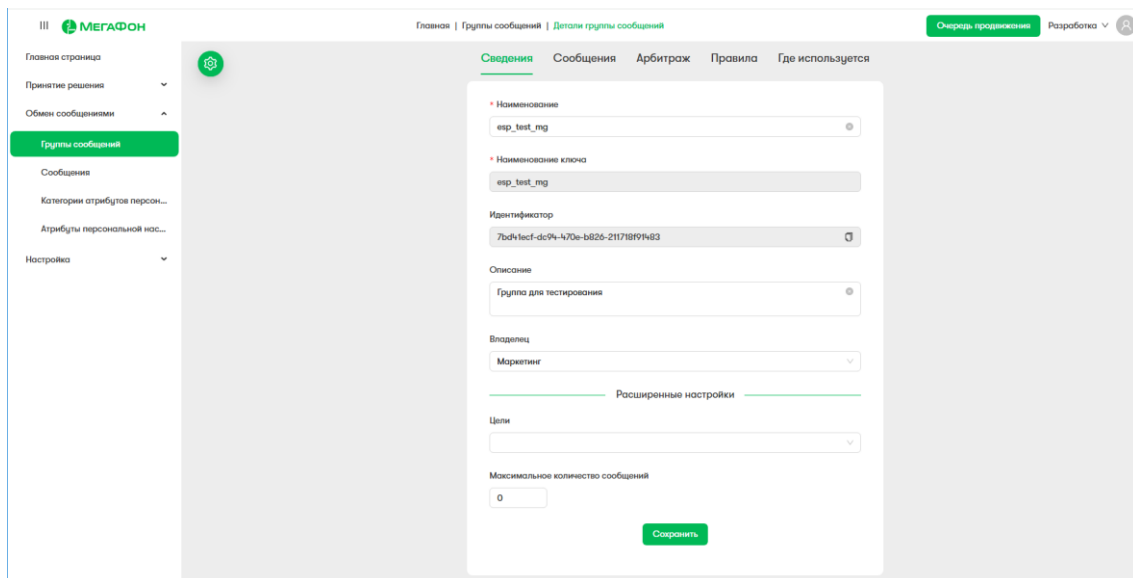


Рисунок 1.10. Группа сообщений

В контексте Группы сообщений представлены табы:

- "Сведения" отображает основную информацию об объекте.

После создания объекта можно навешать Цели для Группы сообщений, задать ограничения на максимальное количество Сообщений подбираемых в группе.

- "Сообщения" список всех Сообщений относящихся к этой Точке взаимодействия.
- "Арбитраж" позволяет задать приоритезацию подобранных Сообщений на уровне Группы сообщений. Не является самостоятельным объектом.
- "Правила" список Правил заданных на объекте.
- "Где используется" список всех Каналов и Точек взаимодействия, с которыми связан данный объект.
- "История изменений" отображает историю добавления/удаления Сообщений в Группу.

## 2.7 Сообщение

Предложение для абонента с настроенными атрибутами персональной настройки, входящими в данное предложение.

Сообщения должны быть связаны с Группами сообщений.

Для создания нового Сообщения необходимо перейти к общему списку всех Сообщений и нажать на кнопку добавления записи. Откроется модальная форма с полями:

- "Наименование" является обязательным для заполнения. Должно быть уникальным среди объектов с одним типом.
- "Наименование ключа" является обязательным для заполнения, заполняется автоматически на основе значения в поле "Наименование", доступно для редактирования в моменте создания, после сохранения объекта значение в поле "Наименование ключа" недоступно для редактирования. Должно быть уникальным среди объектов с одним типом.
- "Описание" не является обязательным для заполнения.
- "Владелец" является обязательным для заполнения.

Мастер-системой для создания предложений является Customer Interaction Manager (далее по тексту СІМ). Там данный объект называется коллатерал. При создании предложения можно выбрать Группы сообщений, в которые оно будет включено и навешать Правила.

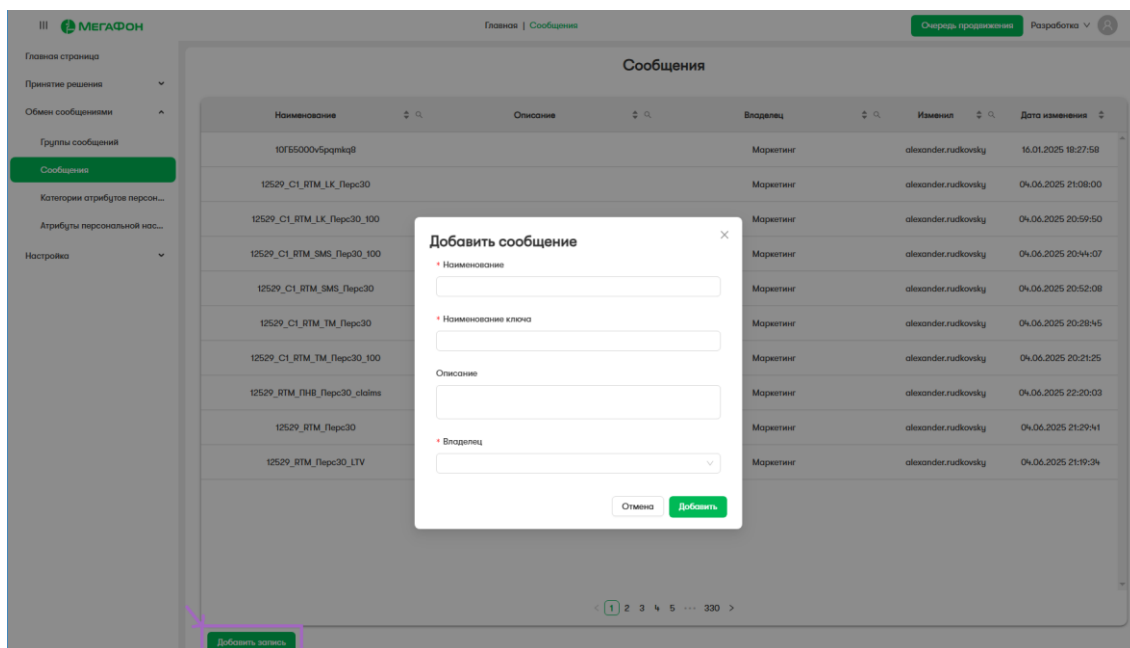


Рисунок 1.11. Создание Сообщения

После добавления записи открывается детальная форма объекта.

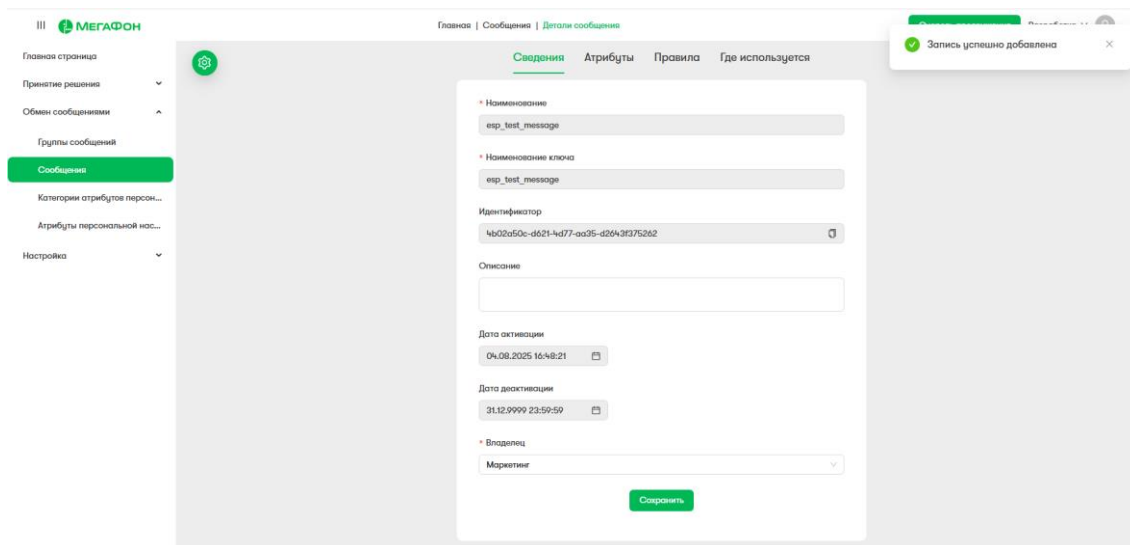


Рисунок 1.12. Сообщение

В контексте Сообщения представлены табы:

- "Сведения" отображает основную информация об объекте.
- "Атрибуты" список Атрибутов персональной настройки, заданных на объекте.
- "Правила" список Правил заданных на объекте.
- "Где используется" список всех Каналов, Точек взаимодействия и Групп сообщений, с которыми связан данный объект.
- "История изменений" отображает историю изменения объекта.

## 2.8 Атрибуты персональной настройки

Атрибуты Сообщения, которые используются в момент формирования предложения для абонентов.

В системе подготовлены атрибуты предложения удовлетворяющие требованиям проводимых кампаний. Следует учитывать, что произвольно созданные атрибуты не будут обработаны каналами коммуникации. При проектировании Каналов и Точек взаимодействия фиксируется ожидаемый набор параметров предложения. На стороне системы отправки сообщения клиенту должен быть реализован маппинг атрибутом MegaRITM на атрибуты Каналов. В контексте Сообщения Атрибуты персональной настройки заполняются автоматически при публикации коллатерала из СИМ на основании маппинга в конфигурационных файлах СИМ-а. Предварительно сам Атрибут должен быть создан в обеих системах.

Для создания нового Атрибута необходимо перейти к общему списку всех Атрибутов персональной настройки и нажать на кнопку добавления записи. Откроется модальная форма с полями:

- "Наименование" является обязательным для заполнения. Должно быть уникальным среди объектов с одним типом.
- "Наименование ключа" является обязательным для заполнения, заполняется автоматически на основе значения в поле "Наименование", доступно для редактирования в моменте создания, после сохранения объекта значение в поле "Наименование ключа" недоступно для редактирования. Должно быть уникальным среди объектов с одним типом.
- "Описание" не является обязательным для заполнения.

- "Категория атрибутов персональной настройки" обязательное для заполнения. Необходимо выбрать значение из выпадающего списка. Заполняется автоматически при вызове формы создания атрибута из контекста Категории атрибутов персональной настройки.
- "Тип значения" обязательное для заполнения. Необходимо выбрать значение из выпадающего списка.
- Переключатель "Стандартный/Справочное/Динамический". Для "Справочных" атрибутов доступно добавление значений для подстановки, при выборе значение может быть только одним. "Динамические" атрибуты служат для определения и подстановки конкретного значения непосредственно в момент подбора Сообщения.
- "Значение по умолчанию"
- Чек-бокс "Разрешить несколько значений"

Рисунок 1.13. Создание Атрибута персональной настройки

После добавления записи открывается детальная форма объекта. Наименование ключа, категория и тип значения атрибута не изменяются после создания.

## 2.9 Категории атрибутов персональной настройки

Объединение Атрибутов персональной настройки в логически связанные группы.

Для создания новой Категории атрибутов необходимо перейти к общему списку всех Категории атрибутов персональной настройки и нажать на кнопку добавления записи. Откроется модальная форма с полями:

- "Наименование" является обязательным для заполнения. Должно быть уникальным среди объектов с одним типом.
- "Наименование ключа" является обязательным для заполнения, заполняется автоматически на основе значения в поле "Наименование", доступно для редактирования в моменте создания, после сохранения объекта значение в поле "Наименование ключа" недоступно для редактирования. Должно быть уникальным среди объектов с одним типом.
- "Описание" не является обязательным для заполнения.

После добавления записи открывается детальная форма объекта. Наименование ключа не изменяется после создания.

## 2.10 Категории атрибутов принятия решения

Объединение Атрибутов, характеризующих абонента в логически связанные группы.

Для создания новой Категории атрибутов необходимо перейти к общему списку всех Категории атрибутов принятия решений и нажать на кнопку добавления записи. Откроется модальная форма с полями:

- "Наименование" является обязательным для заполнения. Должно быть уникальным среди объектов с одним типом.
- "Наименование ключа" является обязательным для заполнения, заполняется автоматически на основе значения в поле "Наименование", доступно для редактирования в моменте создания, после сохранения объекта значение в поле "Наименование ключа" недоступно для редактирования. Должно быть уникальным среди объектов с одним типом.
- "Описание" не является обязательным для заполнения.
- "Владелец" является обязательным для заполнения.

После добавления записи открывается детальная форма объекта. Наименование ключа и владелец не изменяются после создания. Для редактирования доступен чек-бокс "Комплекс" - признак комплексной категории, данные внутри категории представлены в виде массива массивов.

## 2.11 Атрибуты принятия решения

Атрибуты абонентского профиля. Используются в Правилах для задания сегментации при определении наилучшего предложения.

Для создания нового Атрибута необходимо перейти к общему списку всех Атрибутов принятия решения и нажать на кнопку добавления записи. Откроется модальная форма с полями:

- "Наименование" является обязательным для заполнения. Должно быть уникальным среди объектов с одним типом.
- "Наименование ключа" является обязательным для заполнения, заполняется автоматически на основе значения в поле "Наименование", доступно для редактирования в моменте создания, после сохранения объекта значение в поле "Наименование ключа" недоступно для редактирования. Должно быть уникальным среди объектов с одним типом. Регистрозависимо.
- "Описание" не является обязательным для заполнения.
- "Категория атрибута принятия решения" обязательное для заполнения. Необходимо выбрать значение из выпадающего списка. Заполняется автоматически при вызове формы создания атрибута из контекста Категории атрибутов принятия решения.
- "Тип значения" обязательное для заполнения. Необходимо выбрать значение из выпадающего списка.
- Переключатель "Стандартный/Справочное/Динамический". Для "Справочных" атрибутов доступно добавление значений для подстановки, при выборе значение может быть только одним. "Динамический" недоступно для выбора.
- "Значение по умолчанию"
- Чек-бокс "Разрешить несколько значений"

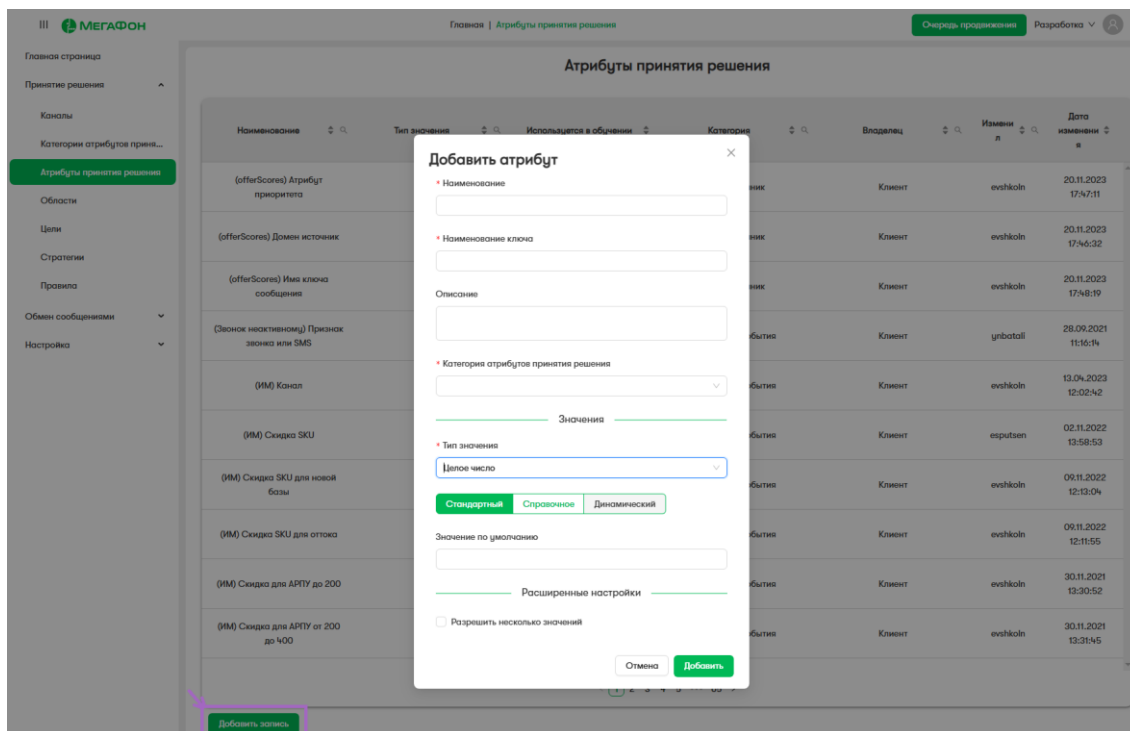


Рисунок 1.14. Создание Атрибута принятия решения

После добавления записи открывается детальная форма объекта. Наименование ключа, категория и тип значения атрибута не изменяются после создания.

## 2.12 Правила

Выражения, позволяющие провести сегментацию абонента по определенному признаку. Например, клиент может воспользоваться предложением А только в том случае, если это физическое лицо, или у его день рождения сегодня.

Для создания нового Правила необходимо перейти к общему списку всех Правил и нажать на кнопку добавления записи. Откроется модальная форма с полями:

- "Наименование" является обязательным для заполнения. Должно быть уникальным среди объектов с одним типом.
- "Наименование ключа" является обязательным для заполнения, заполняется автоматически на основе значения в поле "Наименование", доступно для редактирования в моменте создания, после сохранения объекта значение в поле "Наименование ключа" недоступно для редактирования. Должно быть уникальным среди объектов с одним типом.
- "Описание" не является обязательным для заполнения.
- Чек-бокс "Только сообщения" ограничивает список типов объектов, на которых можно использовать конкретное правило.

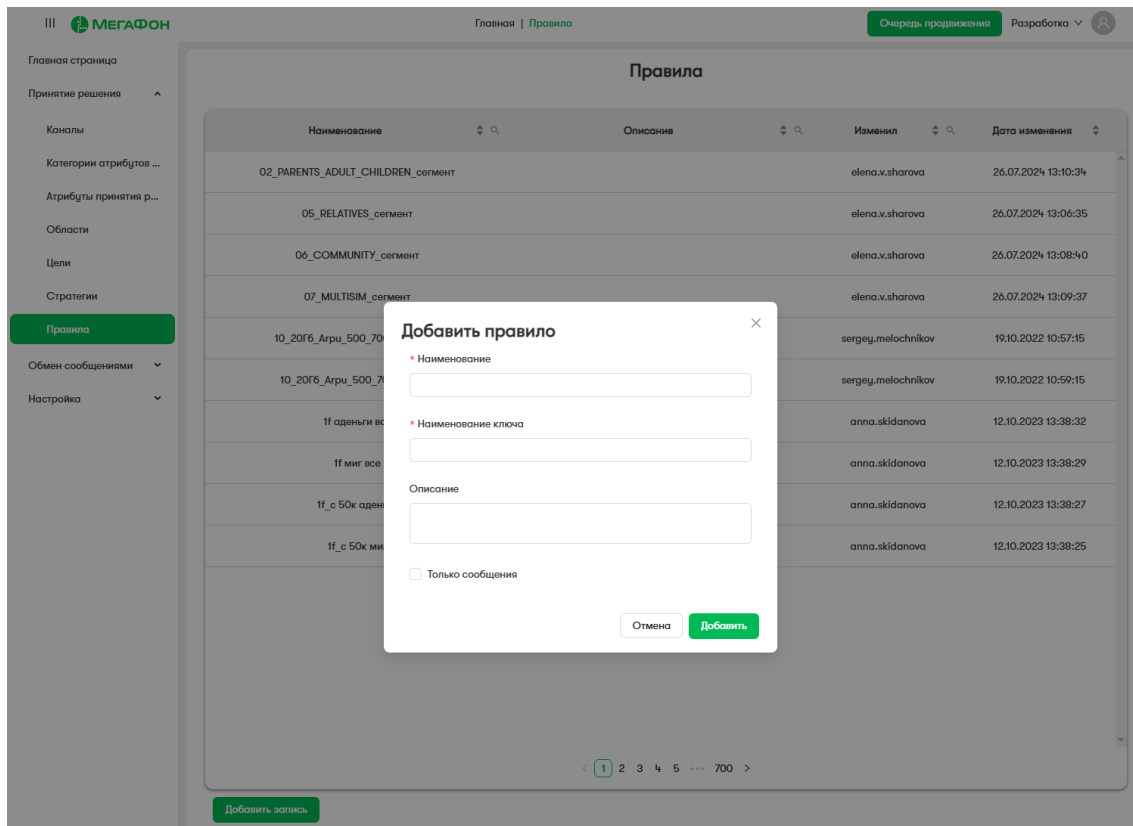


Рисунок 1.15. Создание Правила

После добавления записи открывается детальная форма объекта с табами:

- "Сведения" отображает основную информацию об объекте.
- "Критерии". Тут задаётся само содержание правила.
- "Где используется (контекст)" список всех Каналов, Точек взаимодействия Стратегий и Точек принятия решения, с которыми связан данный объект.
- "Где используется (глобально)" список всех Групп сообщений и Сообщений, с которыми связан данный объект.
- "История изменений" отображает историю изменения объекта.

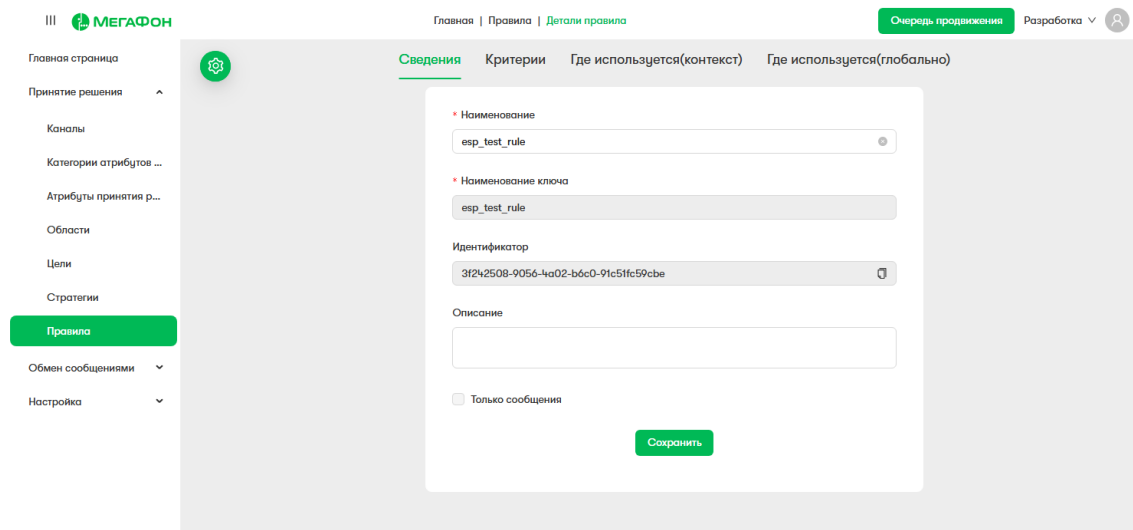


Рисунок 1.16. Правило

Правила могут задаваться на: Канале, Группе сообщений, Сообщении и Стратегии. Для задания необходимо перейти в контекст объекта на табу "Правила", нажать на добавление записи и выбрать правило из списка. И при построении дерева решений привязываться к Точке принятия решения (дочерний объект внутри дерева). Мастер-системой для правил является MegaRITM. Также как и Группы сообщений Правила транслируются в CIM сервисом CDC.

## 2.13 Область

Область - объект настройки деревьев решений, который объединяет следующие за ним объекты по общим параметрам.

На области можно настраивать: Цели, арбитраж (является частью объектов) и Стратегии.

Для создания новой Области необходимо перейти к общему списку всех Областей и нажать на кнопку добавления записи. Откроется модальная форма с полями:

- "Наименование" является обязательным для заполнения. Должно быть уникальным среди объектов с одним типом.
- "Наименование ключа" является обязательным для заполнения, заполняется автоматически на основе значения в поле "Наименование", доступно для редактирования в моменте создания, после сохранения объекта значение в поле "Наименование ключа" недоступно для редактирования. Должно быть уникальным среди объектов с одним типом.
- "Описание" не является обязательным для заполнения.

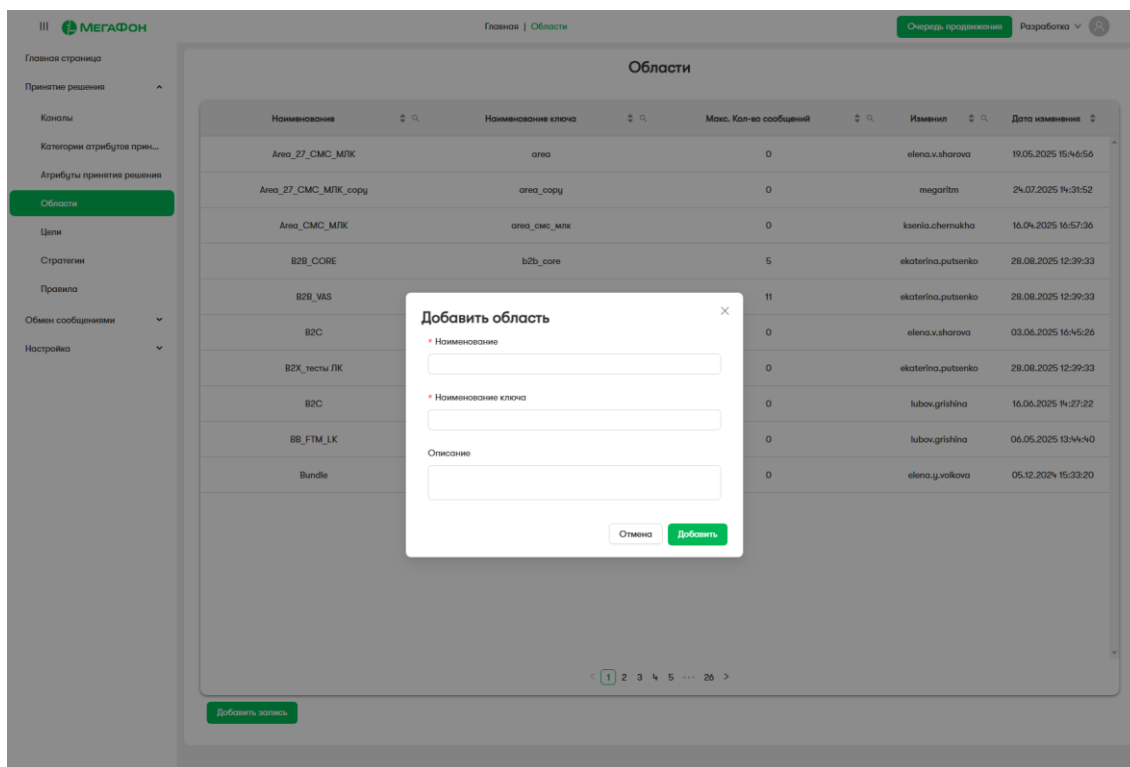


Рисунок 1.17. Создание Области

После добавления записи открывается детальная форма объекта с табами:

- "Сведения" отображает основную информация об объекте. После создания объекта можно навешать Цели для Области, задать ограничения на максимальное количество Сообщений, отбираемых в этой Области.
- "Арбитраж" позволяет задать приоритезацию подобранных Сообщений на уровне Области. Не является самостоятельным объектом.

- "Стратегии" позволяют корректировать оценку сообщения (ранее рассчитанную арбитражем) на основе целей.
- "История изменений" отображает историю изменения объекта.

## 2.14 Арбитраж

Процесс упорядочивания подобранных сообщений по их важности.

Не является самостоятельным объектом. Существует только в контексте:

- Группы сообщений
- Области
- Точки взаимодействия

На всех объектах, доступных для настройки арбитража, на табе "Арбитраж" представлены параметры:

- "Порядок откликов" сортирует предложения в заданном порядке в соответствии с формулой арбитража
- "Формула арбитража" позволяет задать формулу, по которой будет рассчитываться значение арбитража для каждого сообщения. При конфигурировании формулы возможно использовать числовые Атрибуты принятия решения, числовые Атрибуты персональной настройки и знаки арифметических действий.

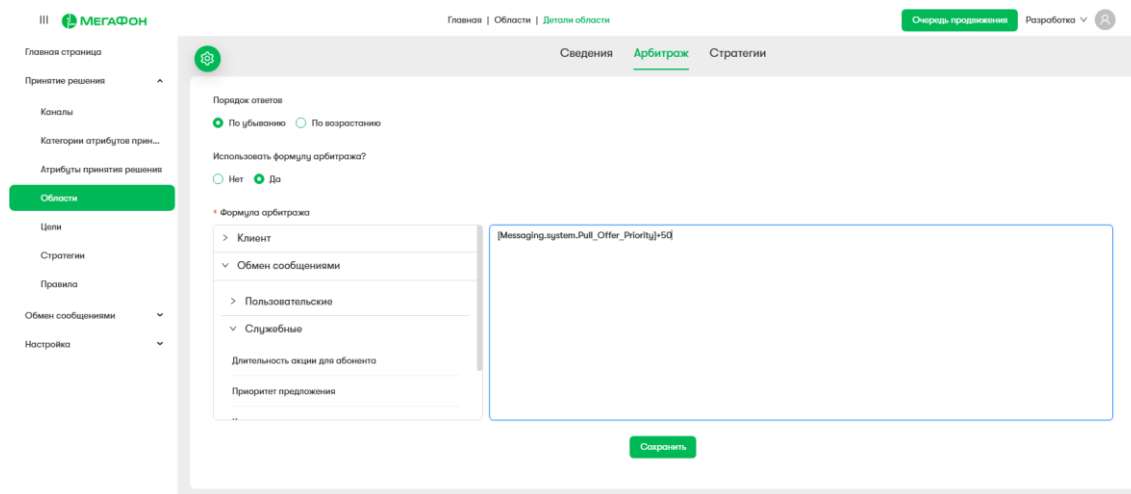


Рисунок 1.18. Арбитраж

В результате применения арбитража все сообщения сортируются по значению арбитража в выбранном порядке откликов и выбираются первые N сообщений, где N – заданное максимальное количество. Максимальное количество задается на табе "Сведения" у всех объектов, где можно задать арбитраж.

## 2.15 Цель

Метка, описывающая бизнес-цель. Используются для арбитража, тестирования и аналитики работы дерева.

Для создания новой Цели необходимо перейти к общему списку всех Целей и нажать на кнопку добавления записи. Откроется модальная форма с полями:

- "Наименование" является обязательным для заполнения. Должно быть уникальным среди объектов с одним типом.
- "Наименование ключа" является обязательным для заполнения, заполняется автоматически на основе значения в поле "Наименование", доступно для редактирования в моменте создания, после сохранения объекта значение в поле "Наименование ключа" недоступно для редактирования. Должно быть уникальным среди объектов с одним типом.
- "Описание" не является обязательным для заполнения.
- "Владелец" является обязательным для заполнения.

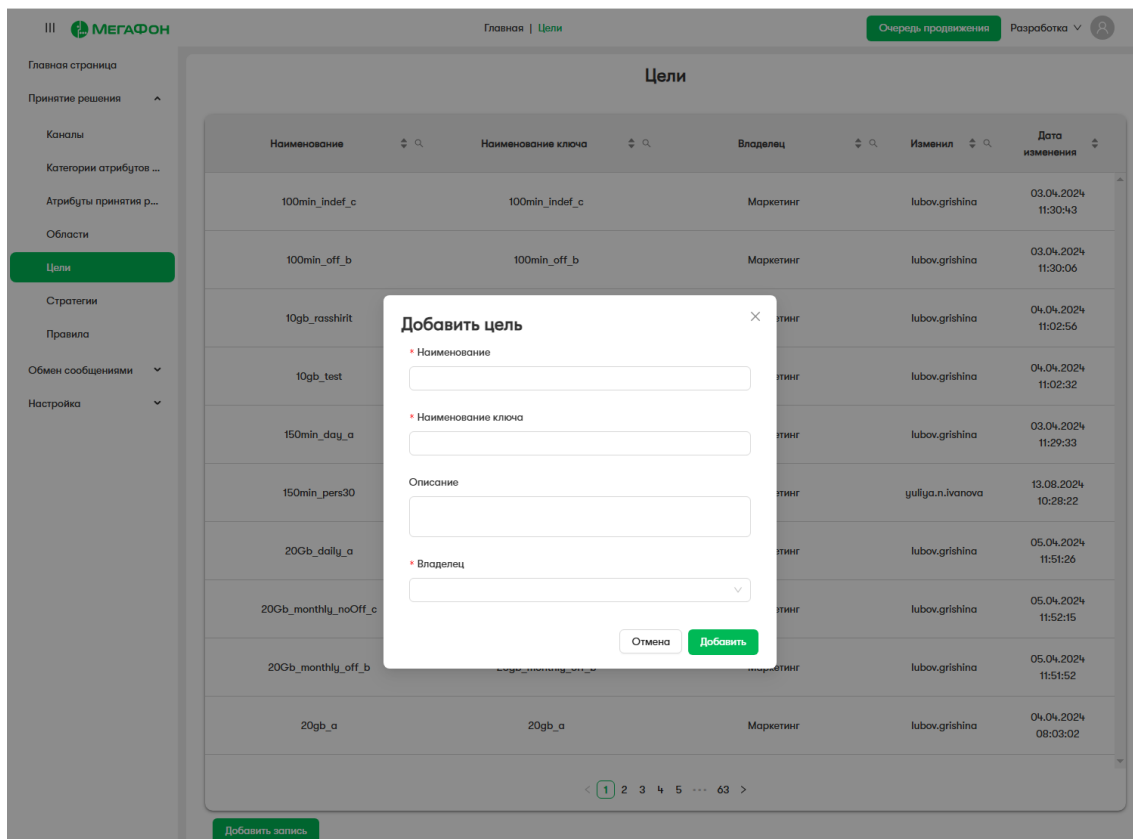


Рисунок 1.19. Создание Цели

После добавления записи открывается детальная форма объекта.

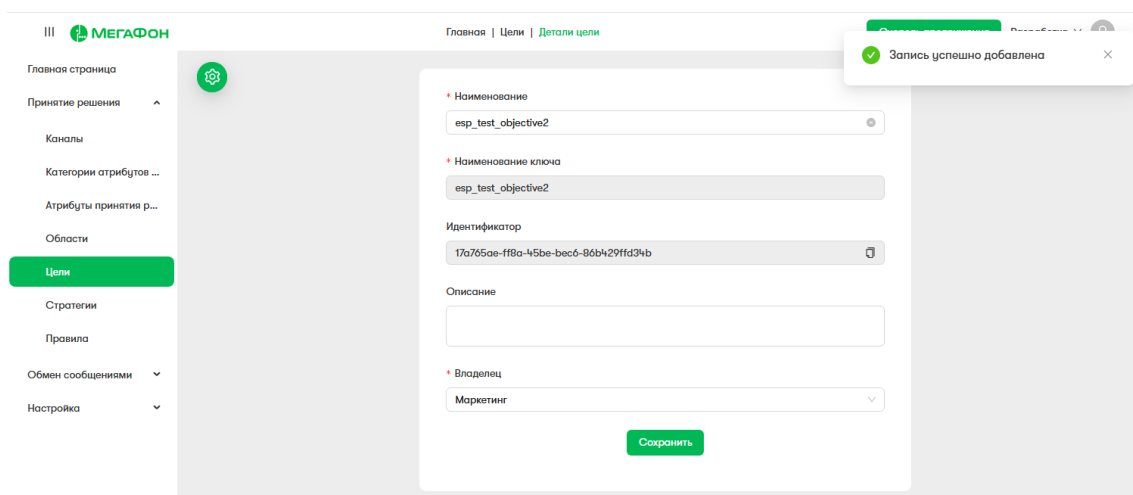


Рисунок 1.20. Цель

Цели можно задавать на: Точке взаимодействия, Области, Группе сообщений. Для задания необходимо перейти в контекст объекта и на табе "Сведения" в поле "Цели" выбрать значение из выпадающего списка.

При прохождении по дереву во время сегментации, все цели, встреченные на пути, суммируются. Таким образом, подобранное Сообщение может иметь разные Цели, которые были заданы на разных объектах дерева решений.

## 2.16 Стратегия

Механизм корректировки оценок Сообщений на основе весов или путём выбора фиксированного количества Сообщений в зависимости от конкретной цели.

Стратегии могут быть заданы на: Точке взаимодействия и Области. Каждая стратегия может быть связана только с одним объектом, при этом объект может иметь 0 - N стратегий. В ситуациях, когда определено несколько стратегий, они указываются в последовательности, и стратегии оцениваются в порядке возрастания. Для создания необходимо перейти в контекст объекта на табу "Стратегии", нажать на добавление записи. Откроется модальная форма с полями:

- "Наименование" является обязательным для заполнения. Должно быть уникальным среди объектов с одним типом.
- "Наименование ключа" является обязательным для заполнения, заполняется автоматически на основе значения в поле "Наименование", доступно для редактирования в моменте создания, после сохранения объекта значение в поле "Наименование ключа" недоступно для редактирования. Должно быть уникальным среди объектов с одним типом.
- "Описание" не является обязательным для заполнения.

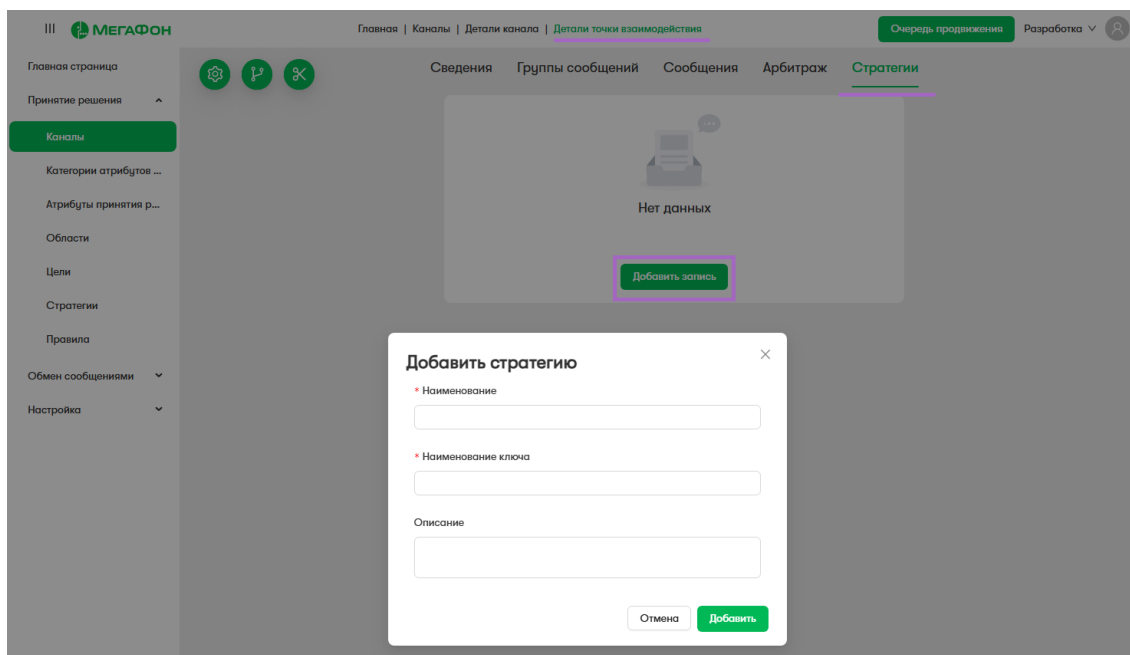


Рисунок 1.21. Создание Стратегии

После добавления записи открывается детальная форма объекта с табами:

- "Сведения" отображает основную информацию об объекте.
- "Балансировщик целей". Тут выполняется настройка весов для всех Целей, которые связаны с объектом.
- "История изменений" отображает историю изменения объекта.

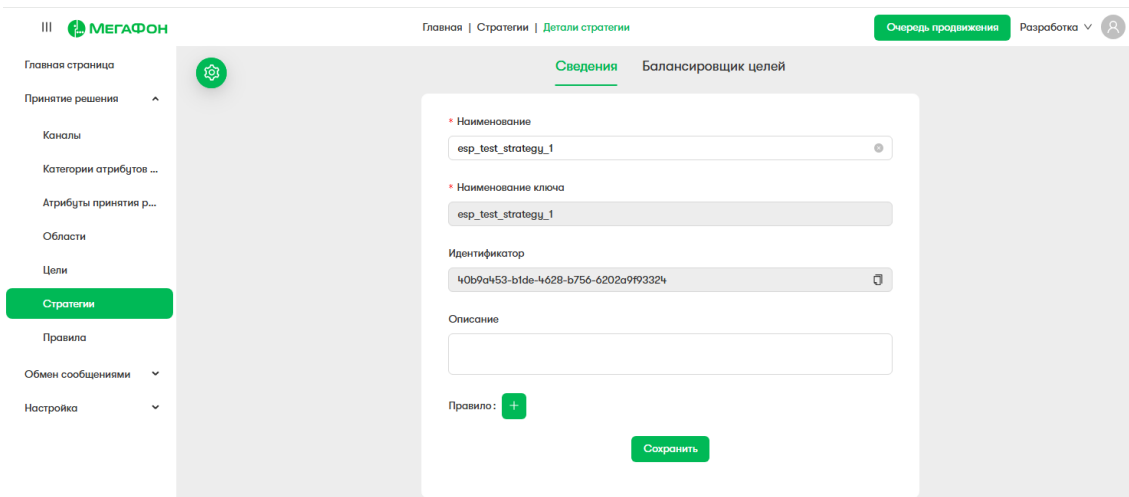


Рисунок 1.22. Стратегия

Стратегии могут иметь Правила, что означает, что стратегия оценивается только при выполнении условия правила.

Возможны два типа балансировки целей: компенсированная(взвешенная) и фиксированная.

При первом открытии все Цели собраны в блоке компенсированных и по умолчанию имеют вес 100%, что означает, что используется полное значение арбитражной оценки для сообщения с соответствующей целью.

Уменьшение веса до 50% приведет к снижению оценки на 50%. Взвешенная оценка 150% добавляет 50% к оценке.

Если сообщение соответствует двум весовым целям, побеждает та, которая набрала больше баллов.

Выбрав чек-бокс в строке в блоке с компенсированными целями, цель перейдет в блок с фиксированными вычислениями.

Этот блок позволяет указывать количество сообщений из каждой цели, которые должны быть возвращены на следующий уровень или в канал. Важна последовательность фиксированных целей подсчета. Когда сообщение относится к двум целям, для сообщения учитывается цель, которая находится первой в последовательности, и это сообщение недоступно для повторного учета для других целей, к которым оно относится.

Если сообщение относится как к фиксированной цели подсчета, так и к взвешенной цели, оно рассматривается как часть фиксированной цели подсчета и возвращается независимо от итоговой оценки сообщения по взвешенной цели.

Примером балансировки оценок Сообщений на основе весов может служить добавление 30% веса ко всем Сообщениям, относящимся к цели «Доход». Примером выбора фиксированного количества Сообщений может быть выбор одного сообщения, относящегося к цели «Удовлетворённость клиентов».

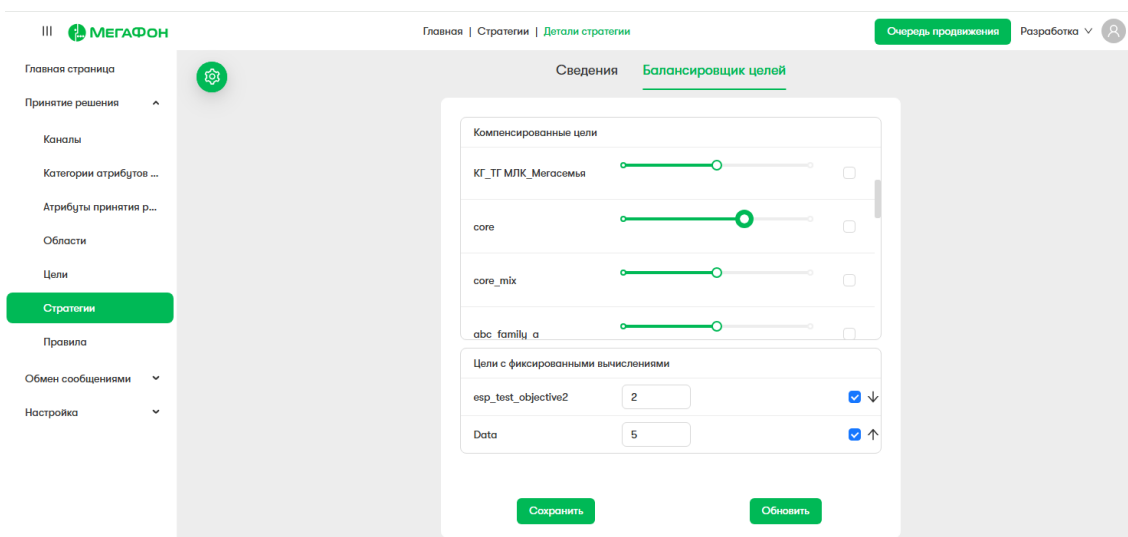


Рисунок 1.23. Балансировщик целей

При добавлении новых Целей к объектам они автоматически не попадают в ранее созданные Стратегии. Для добавления новых целей необходимо нажать на кнопку "Обновить".

## 3 Работа с компонентами

### 3.1 Бизнес среда

После входа в систему вы можете выбрать одну из трёх сред: разработка, тестирование или производство. Элемент для переключения между средами расположен в верхнем правом углу.

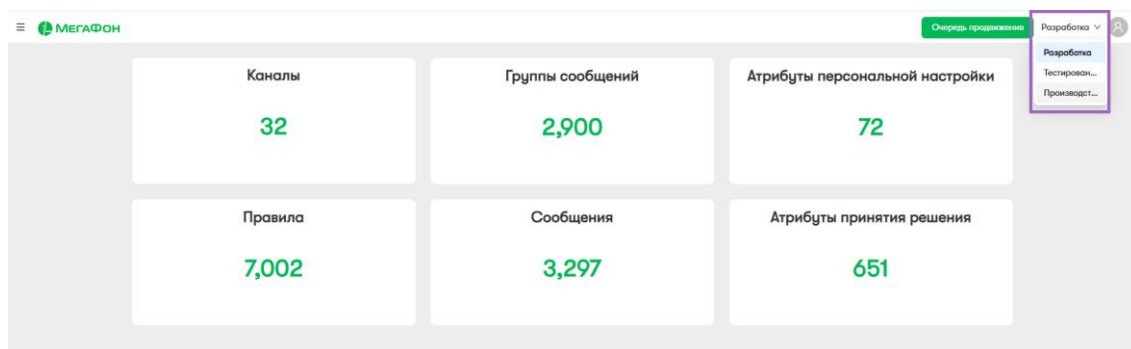


Рисунок 2.1. Переключатель бизнес-сред

По умолчанию выбрана среда "Разработка". Это единственная среда, в которой разрешено редактирование. В среде "Тестирование" выполняется проверка работы деревьев и подбора сообщений без влияния на реальных абонентов. В среде "Производство" происходит подбор сообщений для отправки абонентам в режиме реального времени.

### 3.2 Продвижение компонентов

Перемещение компонентов из одной среды в другую осуществляется через продвижение. Продвижение всегда выполняется последовательно: из среды "Разработка" только в среду "Тестирование" и затем из среды "Тестирование" в "Производство".

Продвижение делится на два типа: одиночное и рекурсивное. Одиночное "Продвижение компонента" - копирует текущее состояние одного объекта из среды-источника в целевую среду. Рекурсивное "Продвижение всех" - запускает копирование корневого объекта и всех его дочерних объектов, которые находятся ниже по иерархии.

Информация по тому какой тип продвижения доступен для объекта представлена в таблице 2.1.

Таблица 2.1 Продвижение компонентов

Тип объекта	Тип продвижения	Результат продвижения
Сообщение	Одиночное	Сообщение и его правила скопированы в целевую среду. Если сообщение входит в группу и эта группа присутствует в целевой среде, то связь тоже будет скопирована.
Группа сообщений	Одиночное	Группа сообщений и её правила скопированы в целевую среду. Если группа содержит сообщения и эти сообщения присутствуют в целевой среде, то связь тоже будет скопирована.
	Рекурсивное	Группа сообщений, её правила, сообщения, правила на сообщениях и связь сообщений с группой скопированы в целевую среду.

Тип объекта	Тип продвижения	Результат продвижения
Правило	Одиночное	Правило скопировано в целевую среду.
Область	Одиночное	Область вместе с заданной на ней формулой арбитража и параметрами сортировки скопированы в целевую среду. Стратегия, заданная на области переносится только при продвижении Точки взаимодействия.
Точка взаимодействия	Рекурсивное	Точка взаимодействия и все объекты, которые однозначно связаны с ней: области, точки принятия решений, группы сообщений и сообщения, иерархически связанные правилами с этими объектами, стратегии скопированы в целевую среду.
Канал	Одиночное	Канал и его правила скопированы в целевую среду.
	Рекурсивное	Канал, его правила и все объекты, которые однозначно связаны с каналом: все точки взаимодействия, их области, точки принятия решений, группы сообщений и сообщения, иерархически связанные правилами с этими объектами, стратегии скопированы в целевую среду.

Объекты с типами:

- Атрибуты принятия решений
- Категории атрибутов принятия решений
- Атрибуты персональной настройки
- Категории атрибутов персональной настройки
- Цели

не требуют продвижения и создаются сразу на всех средах.

Стратегии могут быть созданы на Точке взаимодействия и Области. Список целей используемых в стратегии формируется при построении дерева и собирается из целей, заданных на нижестоящих объектах дерева. Продвижение стратегии происходит только при продвижении точки взаимодействия или при рекурсивном продвижении канала.

Для запуска продвижения необходимо в контексте объекта раскрыть меню действий в верхнем левом углу и выбрать одно из значений: "Продвижение компонента" или "Продвижение всех".

Действие доступно только для пользователей, которым выдано разрешение на продвижение.

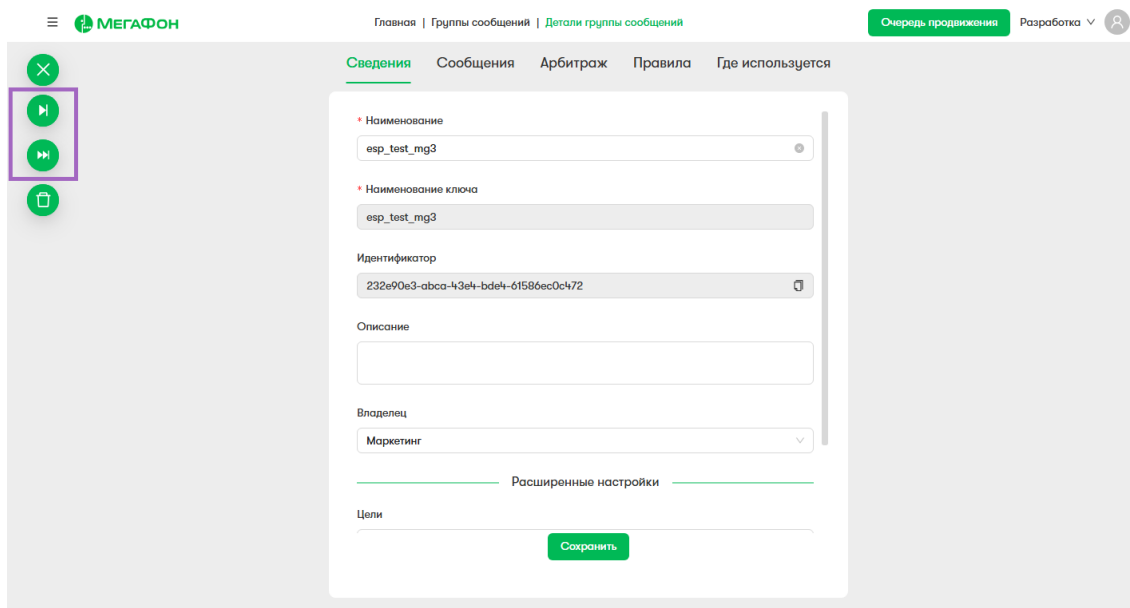


Рисунок 2.2. Продвижение компонента

Задание на продвижение попадает в очередь. Посмотреть список с очередью заданий и статус их выполнения можно в дашборде "Очередь продвижений". Кнопка для перехода к дашборду расположена в верхнем правом углу рядом с указанием среды.

### 3.3 Удаление компонентов

Удаление компонентов имеет ряд ограничений и происходит по-разному в зависимости от типа объекта. Информация представлена в таблице 2.2.

Таблица 2.2. Удаление компонентов

Тип объекта	Ограничение	Результат удаления
Сообщение	Привязка к группам. Для удаления необходимо сначала удалить сообщение из группы.	Объект удален.
Группа сообщений	Привязка к точкам взаимодействия (деревьям). Для удаления группы сообщений предварительно её нужно удалить из всех деревьев, где она используется.	Объект удален. Все сообщения, которые были связаны с группой, остаются в системе без связи с удаленной группой.
Правило	Привязка к узлам дерева, упоминания в группах сообщений, сообщениях, стратегиях. Объект не должен использоваться чтобы его можно было удалить.	Объект удален.
Атрибут	Привязка к правилу (для атрибутов принятия решений) или сообщению (для атрибутов персонализации). Удаление доступно только для администраторов системы.	Объект удален.
Категория атрибутов	Атрибуты группы принятия решения не должны использоваться в правилах.	Объект удален вместе с входящими в группу атрибутами.

Тип объекта	Ограничение	Результат удаления
	Атрибуты группы персонализации не должны использоваться в сообщениях. Удаление доступно только для администраторов системы.	
Цель	Привязка к области, точке взаимодействия или группе сообщений. Объект не должен использоваться чтобы его можно было удалить.	Объект удален.
Стратегия	Из табличного вида нельзя удалять стратегии, привязанные к областям или точкам взаимодействия. Из формы редактирования области или точки взаимодействия стратегия удаляется без предупреждений.	Объект удален.
Область	Привязка к точке взаимодействия (дереву). Для удаления области предварительно её нужно удалить из дерева, где она используется.	Объект удален вместе с использующимися стратегиями.
Точка взаимодействия	Удаление доступно только для администраторов системы.	Объект удален вместе с: точками взаимодействия и стратегиями, которые существуют только в контексте точки взаимодействия. Удаление разрывает связь любых компонентов с точкой взаимодействия.
Канал	Удаление доступно только для администраторов системы.	Объект удален вместе с: точками взаимодействия и их точками принятия решений со стратегиями. Удаление разрывает связь любых компонентов с каналом и его точками взаимодействия, и точками принятия решений.

Удаление объекта доступно только в среде "Разработка" и может быть вызвано из:

1. списочной формы

Для удаления объекта выделите строку с ним и нажмите на кнопку удаления.

2. контекста объекта

Для удаления объекта раскройте блок кнопок в верхнем левом углу и нажмите на кнопку удаления.

### 3.4 Восстановление компонентов

Список удаленных объектов, доступных для восстановления отображается в корзине.

При необходимости удаленные объекты могут быть восстановлены. Для этого необходимо выделить строку с объектом и нажать на кнопку восстановления. Объект будет перенесен из корзины в список объектов с соответствующим типом в среде "Разработка". Объекты восстанавливаются вместе со своей историей изменений.

Типы объектов, которые можно восстановить:

- Правило
- Группа сообщений (без сообщений)
- Область
- Стратегия
- Цель

Восстановление канала доступно только для администраторов системы.

## 4 Работа с правилами

### 4.1 Логика правил

Правила определяют имеет ли клиент право получать определённые сообщения или подходит ли определенное сообщение для данного клиента. Если правила определяют, что клиент не имеет права на получение сообщений, то система не включает эти сообщения в последующую обработку и оценку арбитража.

Примеры использования правил:

Определение соответствия клиента требованиям предложения:

- Ещё не владеет продуктом
- Имеет хорошую кредитную историю

Предотвращение повторных предложений:

- Не отклонял предложение за последние 60 дней

Определение правил гарантирует, что система не будет показывать неподходящие сообщения. Определение правил также уменьшает количество сообщений, которые необходимо обработать при выборе целевых сообщений для клиента. Например, для группы сообщений можно настроить правило, которое проверяет, что возраст клиента не менее 21 года. Если клиент слишком молод, система никогда не покажет клиенту ни одно из сообщений, принадлежащих этой группе сообщений.

Добавление слишком большого количества правил может исключить подходящие сообщения, тем самым мешая MegaRITM выбирать оптимальное сообщение для клиента. Связывание конфликтующих правил (например, связывание одного правила с группой сообщений, требующей, чтобы клиент был моложе 21 года, и связывание другого правила, требующего, чтобы клиент был старше 21 года) может помешать отображать какие-либо сообщения вообще.

### 4.2 Компоненты, которые могут иметь правила

Правила могут задаваться на:

- Канале,
- Области,
- Группе сообщений,
- Сообщении,
- Стратегии.

Для задания необходимо перейти в контекст объекта на табу "Правила", нажать на добавление записи и выбрать правило из списка. И при построении дерева решений привязываться к

Кроме того, правила можно указывать при построении дерева, на точках принятия решения. Например, если правило относится к пути, ведущему к группе сообщений, оно применяется к этой группе сообщений только при следовании по этому пути. Если правило связано с самой группой сообщений, оно применяется везде, где используется эта группа сообщений.

## 4.3 Конструктор правил

MegaRITM предоставляет интерфейс с функцией перетаскивания компонентов для конфигурирования правил.

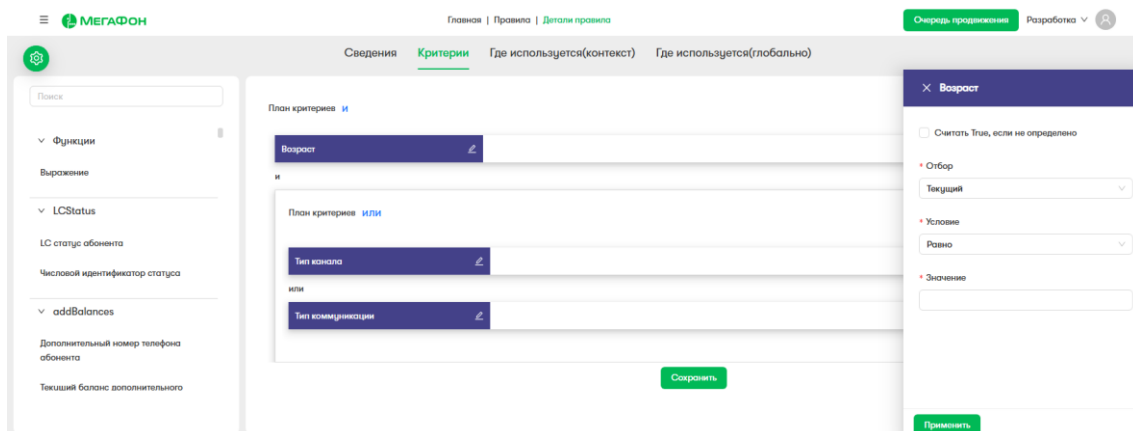


Рисунок 3.1. Конструктор правил

Слева на форме расположено основное меню компонентов. В первой строке всегда закреплены функции. Ниже список атрибутов принятия решения, объединенный по их категориям. Доступен поиск по части наименования атрибута.

В центральную часть на рисунке 2.1 добавлено несколько атрибутов принятия решения. Правило может содержать один или несколько критериев. Критерии можно объединять в блоки. Оператор связи между критериями по умолчанию - "И", возможно изменение на "ИЛИ".

При нажатии на иконку редактирования на компоненте, справа открывается сайт-бар для задания параметров. Параметры различаются в зависимости от типа атрибута и его группы. Например, атрибут, представляющий собой строку, имеет параметры, отличающиеся от параметров целого числа или логического значения. Функции также имеют параметры, отличающиеся друг от друга.

## 4.4 Критерии отбора и условия

Параметр «Считать True, если не определено» всегда доступен на панели редактирования для атрибутов (но не для функций). Вы можете отметить чек-бокс, если информация о значениях атрибутов, необходимых для правила, может быть недоступна при оценке правила.

Для атрибутов, чьи категории не являются комплексными, значение отбора всегда будет текущим.

Для атрибутов из комплексных категорий отбор может принимать значения:

- Все - все значения атрибута должны соответствовать условиям,
- По крайней мере 1 из - одно из значений атрибута должно соответствовать условиям,
- Сумма - сумма значений атрибутов должна соответствовать условиям. Применимо только для числовых типов атрибутов.

Для задания условий проверки числовых атрибутов доступны следующие значения:

- Равно,
- Не равно,
- Меньше,

- Меньше или равно,
- Больше,
- Больше или равно,
- Между,
- Значение пусто,
- Значение не пусто.

Для задания условий проверки строковых атрибутов доступны следующие значения:

- Равно,
- Не равно,
- Начинается с,
- Не Начинается с,
- Заканчивается на,
- Не заканчивается на,
- Содержит,
- Не содержит,
- Совпадает с,
- Не совпадает с,
- Значение пусто,
- Значение не пусто.

Для логических атрибутов условия могут принимать значения:

- Равно,
- Не равно.

Для атрибутов с типом дата можно указывать:

- Фиксированную дату или диапазон
- Относительный диапазон

## 4.5 Настройка правил

### 4.5.1 Правило с несколькими атрибутами и объединением атрибутов в блоки

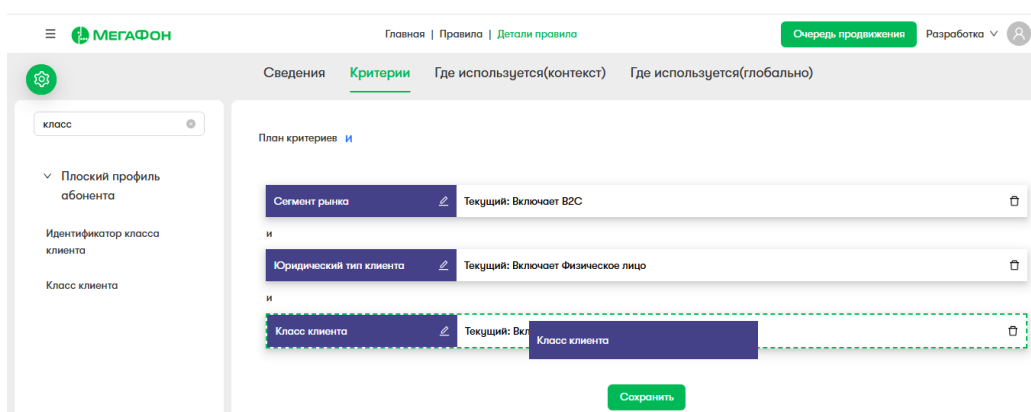
Создадим правило для проверки базовых условий, таких как: сегмент рынка, юридический тип клиента, класс клиента и срок жизни абонента.

Открыт контекст правила, таб "Критерии" в среде "Разработка".

1. Перетащите из меню компонентов слева атрибут "Сегмент рынка" в поле с планом критериев. Это атрибут со значениями для подстановки.
2. Нажмите на иконку редактирования компонента и задайте параметры: Отбор = Текущий, Условие = Включает, Значение выберете из выпадающего списка "B2C".
3. Нажмите кнопку "Применить".
4. Перетащите из меню компонентов слева атрибут "Юридический тип клиента" в поле с планом критериев ниже первого атрибута. Это атрибут со значениями для

подстановки. Для удобства можно воспользоваться поиском атрибута по его наименованию.

5. Нажмите на иконку редактирования компонента и задайте параметры: Отбор = Текущий, Условие = Включает, Значение выберите из выпадающего списка "Физическое лицо".
6. Нажмите кнопку "Применить".
7. Добавьте атрибут "Класс клиента" из меню компонентов. Это атрибут со значениями для подстановки.
8. Нажмите на иконку редактирования компонента и задайте параметры: Отбор = Текущий, Условие = Включает, в поле Значение выберите несколько значений из выпадающего списка "Базовый", "Нормальный".
9. Нажмите кнопку "Применить".
10. Добавьте ещё раз атрибут "Класс клиента" из меню компонентов в один блок с первым использованием "Класса клиента". Для этого зажмите компонент и перенесите его в поле с планом критериев, так чтобы предыдущий компонент начал подсвечиваться.



11. Нажмите на иконку редактирования компонента и задайте параметры: Отбор = Текущий, Условие = Содержит, в поле Значение введите "Золотой".
12. Нажмите кнопку "Применить".
13. Измените оператор связи для компонентов в блоке на "ИЛИ".
14. Добавьте атрибут "Класс клиента" из меню компонентов.
15. Нажмите на иконку редактирования компонента и задайте параметры: Отбор = Текущий, Условие = Больше, в поле Значение введите "91".
16. Нажмите кнопку "Применить".
17. Нажмите кнопку "Сохранить".

Правило сконфигурировано. Проверка будет пройдена только если будут выполнены все условия: абонент относится к сегменту B2C, является физическим лицом, его класс "Базовый" или "Нормальный" или любой вариант содержащий подстроку "Золотой", например "Золотой\_2024" и срок жизни абонента больше 91 дня.

#### 4.5.2 Правило с датами

Создадим правило для проверки попадания абонента в блокировку в течение последних двух недель независимо от конкретных дат. Открыт контекст правила, таб "Критерии" в среде "Разработка".

1. Перетащите из меню компонентов слева атрибут "Дата входа в блокировку" в поле с планом критериев.
2. Нажмите на иконку редактирования компонента и задайте параметры: Отбор = По крайней мере 1 из. Это атрибут комплексной категории.
3. Нажмите кнопку "Добавить дату".
4. В открывшемся модальном окне переключите "Тип даты" на "Относительную".
5. Задайте значение продолжительности = 14 дням.
6. Задайте значение для начала периода = 14 дням.
7. Переключатель "Диапазон даты относительно даты запуска" оставьте в значении "В прошлом"
8. Нажмите кнопку "Сохранить".
9. Нажмите кнопку "Применить".
10. Нажмите кнопку "Сохранить".

Правило сконфигурировано. Абонент мог несколько раз входить в блокировку. Проверка будет пройдена если хотя бы одна из дат будет в течение последних 14 дней.

### 4.5.3 Правило с выражением

Выражение - это функция, которая может быть включена в критерии правила как и любой атрибут принятия решения. Выражение позволяет задать произвольные критерии, которые недоступны при работе с атрибутами.

Справочник с функциями, которые можно использовать в выражениях приведен в Приложении 1

Для примера создадим правило проверяющее историю контактов абонента на наличие просмотров предложения.

Открыт контекст правила, таб "Критерии" в среде "Разработка".

1. Перетащите из меню компонентов слева "Выражение".
2. Нажмите на иконку редактирования компонента и в открывшемся справа сайдбаре введите выражение:

```
conditional(get("Messaging.system.CIM_Collateral_Id"), NaN, false)
&&
conditional(get("Customer.relevantOffers.collateralIdsActive"),
NaN, false)
&&
pattern( ".*"+get("Messaging.system.CIM_Collateral_Id")+".*", 2,
get("Customer.relevantOffers.collateralIdsActive") )
&&
conditional(get("Customer.relevantOffers.collateralIdsAccepted"),
NaN, true)
&&
!pattern( ".*"+get("Messaging.system.CIM_Collateral_Id")+".*", 2,
get("Customer.relevantOffers.collateralIdsAccepted") )
```

3. Нажмите кнопку "Применить".
4. Нажмите кнопку "Сохранить".

Правило сконфигурировано. При использовании его на Сообщении будет выполняться:

- определение идентификатора текущего сообщения
- определение идентификаторов актуальных предложений из истории контактов абонента в разрезе канала
- поиск идентификатора текущего сообщения среди актуальных предложений

- определение идентификаторов предложений по которым есть отклики второго или третьего типа
- поиск идентификатора текущего сообщения среди идентификаторов предложений с откликами

Правило будет пройдено в случае, если среди актуальных предложений для абонента присутствует данное сообщение и по нему отсутствуют отклики второго и третьего типов.

## 5 Настройка деревьев

### 5.1 Переход к графическому режиму

Для подбора предложения абоненту, недостаточно иметь созданные Канал, Точку взаимодействия, Группу сообщений и Сообщение. Объекты должны быть связаны друг с другом. Для выстраивания связей и построения дерева решений необходимо открыть Точку взаимодействия в графическом режиме.

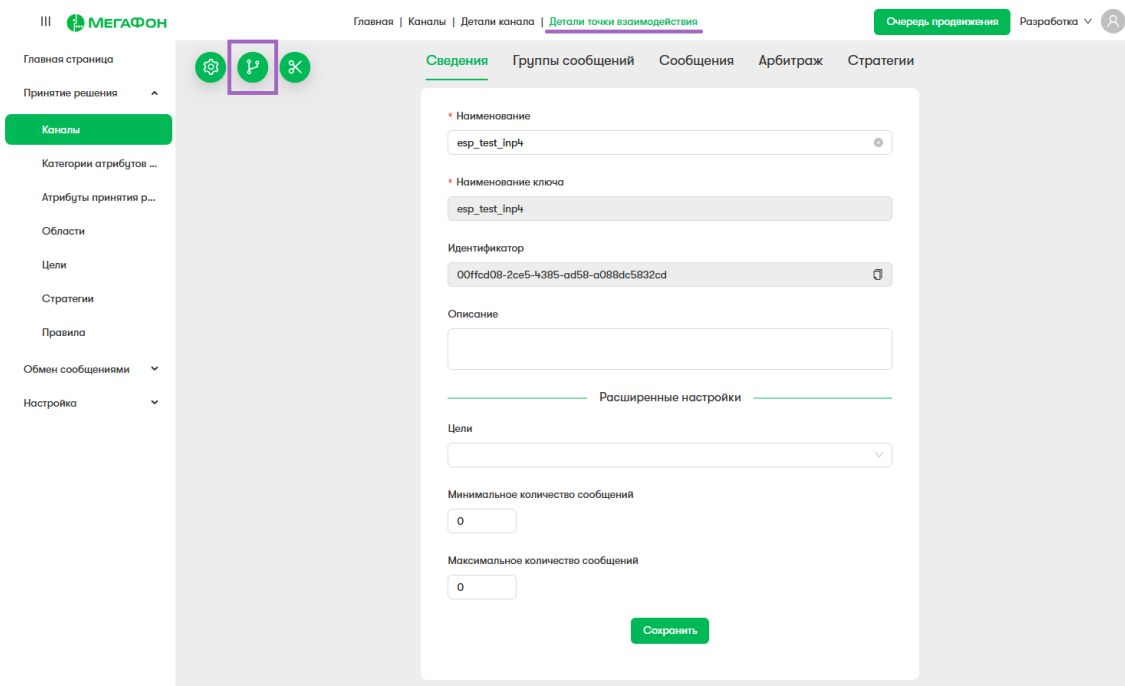


Рисунок 4.1 Переход к графическому режиму

Просматривать деревья решений можно в любой среде, редактирование доступно только в среде разработки.

### 5.2 Работа в редакторе дерева

После открытия графического режима Точка взаимодействия будет единственным компонентом на доске.

Для перехода к редактированию нажмите на иконку "карандашика" в верхнем левом углу.

Редактор дерева доступен для совместной работы нескольких пользователей. Поэтому графический вид дерева в режимах чтения и редактирования может отличаться друг от друга. Все изменения вносимые одним пользователем отображаются у другого пользователя в режиме реального времени.

Кликните по компоненту дерева для отображения сайдбара. На этой панели вы можете:

- Добавлять пункты назначения к компоненту
- Редактировать свойства компонента
- Удалять компонент

Используйте колёсико мыши для изменения размера на доске. Для перемещения по доске зажмите левую кнопку мыши и переместите её.

## 5.3 Добавление пунктов назначения

Вы можете добавить пункты назначения к Точке взаимодействия. Для выбора доступны:

- Область - откроется поп-ап окно со списком Областей, которые не задействованы в деревьях решений.
- Точка принятия решения - откроется модальная форма создания точки принятия решения. Существует только внутри иерархии дерева, недоступна для переиспользования.
- Группа сообщений - откроется поп-ап окно со списком всех созданных Групп сообщений.

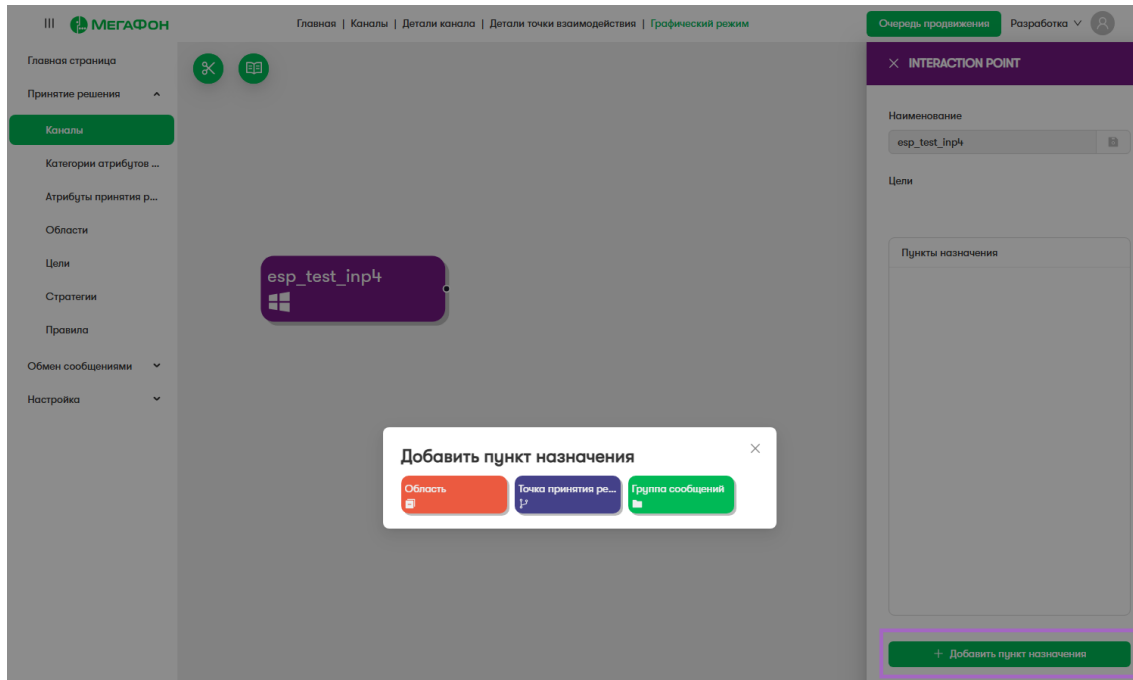


Рисунок 4.2 Добавление пунктов назначения

Группы сообщений находятся на самом нижнем уровне иерархии доски, поэтому к ним нет пунктов назначения, которые можно было бы добавить.

## 5.4 Удаление узлов

Вы можете удалить любую Точку принятия решения, Область или Группу сообщений из иерархии дерева. Дочерние пункты назначения удаленного пункта назначения также удаляются.

Удаление Группы сообщений удаляет только ссылку на группу сообщений из дерева. Группа сообщений не удаляется из системы.

Удалить Точку взаимодействия из дерева невозможно.

## 5.5 Редактирование дерева

Смена источника, смена порядка веток, навешивание правил (+ логика включая/исключая)

### 5.5.1 Добавление правил

Для определения какое из сообщений больше подходит конкретному абоненту на переходы к узлам можно прикреплять правила. Выделите узел дерева от которого есть переход к последующим узлам и в сайтбаре выберите "Добавить правило". Откроется модальное окно со списком всех созданных в системе правил. Выберите правило из списка.

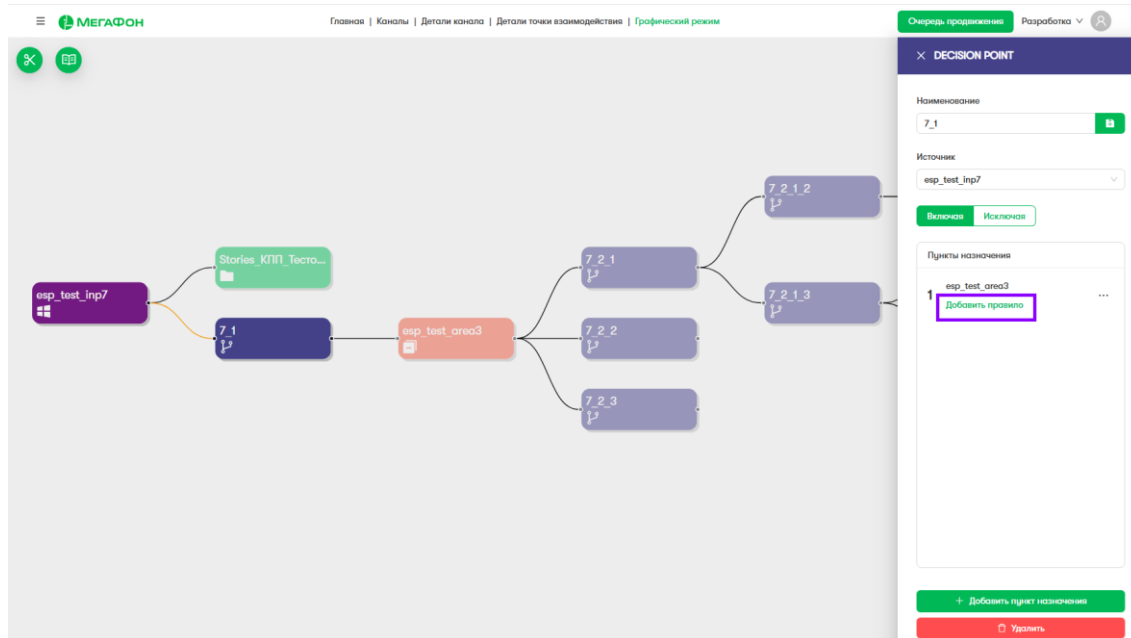


Рисунок 4.3. Добавить правило

Правила можно задавать на:

- Точке взаимодействия
- Точке принятия решения

На переход к следующему узлу можно добавить только одно правило. Для удаления прикрепленного правила в сайтбаре справа необходимо нажать на три точки рядом с правилом и выбрать "Удалить правило". Это действие удаляет связь правила с деревом, само правило останется в системе.

### 5.5.2 Логика управления потоком

Через проверку правил можно управлять сегментацией абонентов и направлять их в нужную ветку ведущую к наиболее подходящему сообщению. По умолчанию все новые узлы добавляются с переключателем в положении "Включая". Это значит что один запрос будет параллельно направлен во все ветки выходящие из узла. При выборе значения "Исключая" запрос будет последовательно направлен сначала в первую ветку. Если проверка правила будет пройдена успешно, то запрос продолжит движение только по ней и не попадет во вторую и все последующие ветки. Если проверка не будет пройдена, то запрос будет направлен во вторую ветку.

### 5.5.3 Изменение порядка веток

При использовании логики "Исключая" важна очередность веток, ведь проверка выполняется последовательно. Изменить порядок можно при помощи стрелок в сайтбаре. Нажимая на стрелку вверх или вниз узел со всеми следующими узлами будет перемещен выше или ниже на поле с деревом.

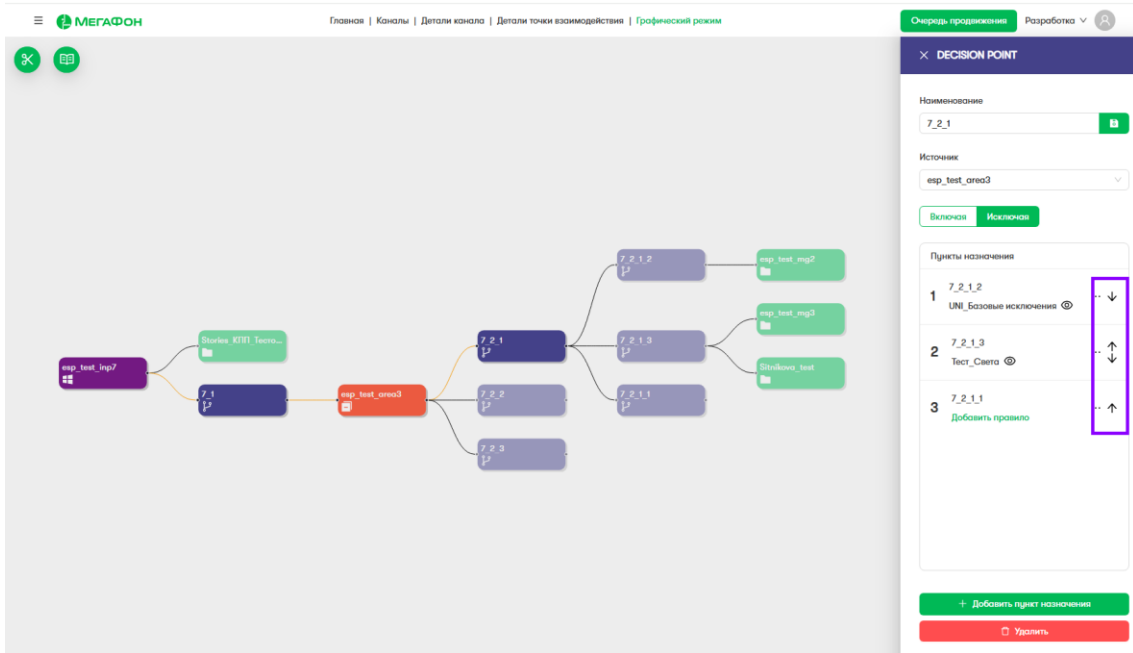


Рисунок 4.4. Смена порядка веток

### 5.5.4 Изменение источника

Для того чтобы не просто изменить порядок веток а полностью перенести ветку в другое место на дереве достаточно раскрыть выпадающий список значений в поле "Источник" и выбрать необходимый узел. На рисунке 4.5 ветка 7.2.1 перенесена с области "esp\_test\_area3" к точке принятия решения "7\_1". Предыдущее состояние показано на рисунке 4.4.

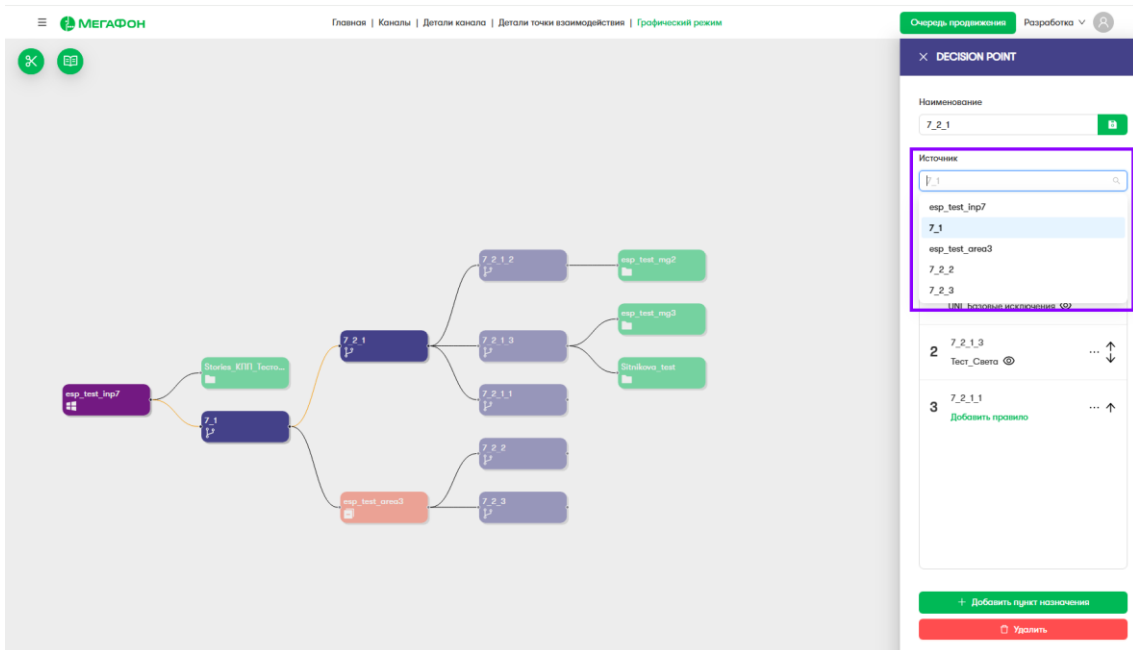


Рисунок 4.5. Изменение источника

### 5.6 Сохранение изменений

Для сохранения изменений в дереве необходимо нажать на кнопку сохранения в верхнем правом углу и подтвердить своё действие.

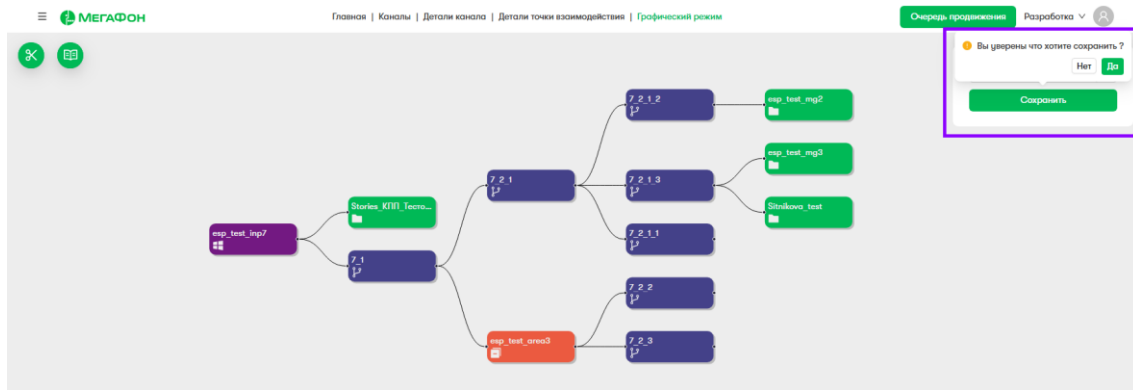


Рисунок 4.6. Сохранение изменений

Для выхода без сохранения достаточно перейти в режим чтения или на любую форму системы.

При работе нескольких пользователей в режиме редактирования сохраняются все внесенные в дерево изменения. При выходе без сохранения, внесенные изменения будут доступны в режиме редактирования до тех пор, пока все пользователи не выйдут из режима редактирования.

## 5.7 Дополнительные настройки

Дополнительными настройками можно считать любые параметры влияющие на результат подбора сообщения, которые не требуют настройки в графическом режиме.

Правила можно задать не только в графическом режиме на переходах между узлами дерева, но и на:

- Сообщении
- Группе сообщений
- Стратегии
- Канале

Максимальное количество сообщений можно задать на:

- Группе сообщений
- Области
- Точке взаимодействия

Минимальное количество сообщений можно задать на:

- Точке взаимодействия

Цели можно задать на:

- Группе сообщений
- Области
- Точке взаимодействия

Арбитраж можно задать на:

- Группе сообщений
- Области
- Точки взаимодействия

Стратегии можно задать на:

- Области
- Точке взаимодействия

## 6 Механизм принятия решения

Подбор сообщения для абонента можно разделить на несколько этапов.

### 6.1 Проверка правил

Первым этапом выполняется проверка профиля абонента на соответствие правилам.

Применение правил иерархично:

- Канал
- Узлы дерева
- Группа сообщений
- Сообщение

Как только правило не выполняется проверка обрывается.

После проверки правил получаем набор предварительно отобранных сообщений. Для каждого сообщения собирается набор целей, которые были заданы на узлах по мере прохождения дерева.

### 6.2 Арбитраж

Вторым этапом к списку предварительно отобранных сообщений применяется ранжирование - определение важности и сортировка в заданном порядке.

В первую очередь обрабатывает арбитраж на уровне Группы сообщений, с выбором максимального количества сообщений на уровне группы; затем – на уровне Области, предыдущие оценки на уровне групп сообщений отбрасываются;

затем – на уровне Точки взаимодействия, предыдущие оценки на уровне групп сообщений и областей отбрасываются.

При этом, для каждого уровня возможно настроить свою формулу арбитража. Оценки, полученные в результате арбитража родительского объекта, переопределяют оценки, полученные в результате арбитража дочернего объекта. Если арбитраж для родителя не определен, оценки рассчитанные на уровне дочернего элемента, сохраняются.

Применяется только к предложениям, которые доступны для подбора абоненту после его прохождения по дереву.

Если арбитраж не задан, то предложения отдаются в случайном порядке.

Если сообщения из разных групп имеют одинаковое значение параметра, по которому выполняется арбитраж и выставлено ограничение на количество сообщений в канале, то будет выдано то, чья группа сообщений пришла в дерево первой, согласно очередности веток в дереве.

### 6.3 Стратегии

Финальным этапом к списку предварительно отобранных сообщений и при наличии арбитража с переопределенным весом и отсортированным в заданном порядке применяется корректировки оценок на основе весов или путём выбора фиксированного количества сообщений с конкретной целью.

Стратегии применяются последовательно от наименьшего звена к наивысшему (справа налево).

Стратегии могут иметь правила, что означает, что стратегия оценивается только при выполнении условия правила.

Возможны два типа балансировки целей: компенсированная(взвешенная) и фиксированная.

Компенсированная изменит вес сообщения (определенный при арбитраже) с определенной целью в соответствии со значением модификатора,

Если сообщение соответствует двум компенсированным целям, побеждает та, которая набрала больше баллов.

Фиксированные вычисления ограничивают количество сообщений из каждой цели, которые должны быть возвращены на следующий уровень или в канал. Важна последовательность фиксированных целей подсчета. Когда сообщение относится к двум целям, для сообщения учитывается цель, которая находится первой в последовательности, и это сообщение недоступно для повторного учета для других целей, к которым оно относится.

Если сообщение относится как к фиксированной цели подсчета, так и к взвешенной цели, оно рассматривается как часть фиксированной цели подсчета и возвращается независимо от итоговой оценки сообщения по взвешенной цели.

## **6.4 Ограничение количества сообщений**

Максимальное количество сообщений ограничивается последовательно от наименьшего звена к наивысшему (справа налево). Сначала на группе сообщений, затем на областях от ближайшей к группе до ближайшей к точке взаимодействия, и наконец на точке взаимодействия.

## 7 Сервис отсечений

### 7.1 Назначение

Сервис отсечений предназначен для анализа эффективности маркетинговых кампаний. Позволяет выявлять узкие места в дереве принятия решений. Дает возможность оценить входящий поток абонентов и процент отсечений на каждом этапе.

### 7.2 Открытие сервиса отсечений

Переход к отсечениям доступен из контекста Точки взаимодействия.

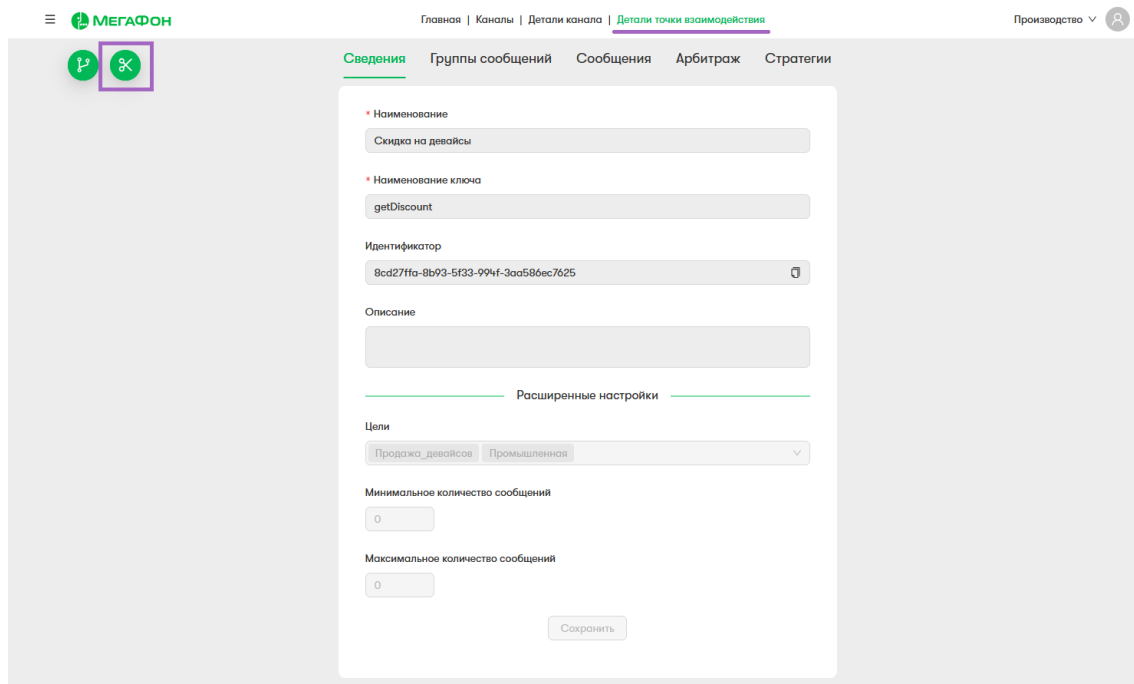


Рисунок 6.1. Переход к сервису отсечений

Сервис отсечений отображает текущее состояние дерева в графическом виде. На каждом узле отображается:

- количество абонентов зашедших в узел
- количество абонентов не прошедших проверку в данном узле

При нажатии на узел открывается сайтбар с детализацией отсечений до каждого правила. При нажатии на Группу сообщений в сайтбаре выводится информация по Сообщениям. Дополнительно ведется подсчет выдачи Сообщения. Это значение может отличаться от количества подборов если на дереве настроен арбитраж и Стратегии.

Подсчет отсечений ведется независимо для сред Тестирования и Производство.



## 8 Администрирование

Функционал в блоке "Настройка" из основного меню навигации доступен только для администраторов системы.

### 8.1 Добавление пользователей

В системе используется интеграция с Active Directory (далее по тексту AD).

Для получения доступа пользователь должен быть добавлен в одну из групп AD:

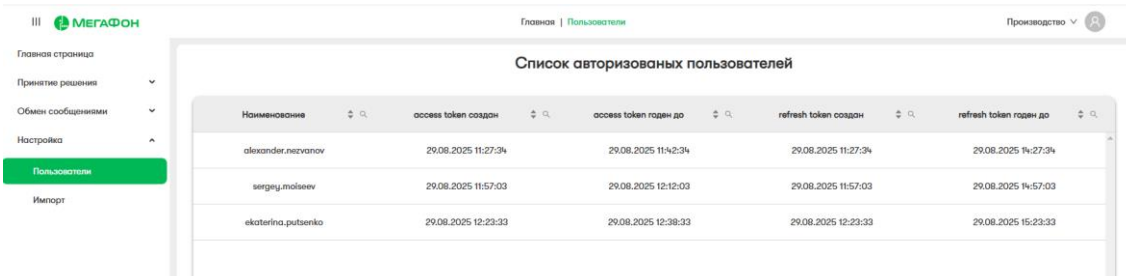
- glb-trm-mr-viewer - группа пользователей без права на редактирование объектов,
- glb-trm-mr-editor - группа пользователей с возможностью редактирования объектов,
- glb-trm-mr-admin - группа пользователей-администраторов системы.

### 8.2 Просмотр авторизованных пользователей

В системы реализована возможность просмотра списка пользователей, которые авторизованы в системе.

Перейти к списку можно из основного меню навигации через блок "Настройка" и затем "Пользователи".

В списке отображается логин пользователя и информация о времени получения и истечения токенов. Список формируется от момента запуска контейнера с сервисом авторизации.



Наименование	access token создан	access token годен до	refresh token создан	refresh token годен до
alexander.nezvanov	29.08.2025 11:27:34	29.08.2025 11:42:34	29.08.2025 11:27:34	29.08.2025 14:27:34
sergey.moliseev	29.08.2025 11:57:03	29.08.2025 12:12:03	29.08.2025 11:57:03	29.08.2025 14:57:03
ekaterina.putsenko	29.08.2025 12:23:33	29.08.2025 12:38:33	29.08.2025 12:23:33	29.08.2025 15:23:33

Рисунок 7.1 Список авторизованных пользователей

### 8.3 Импорт данных

В MegaRITM возможен перенос данных из системы Real-Time Interactin Manager (далее по тексту RTIM).

Данные из RTIM должны быть предоставлены в виде двух файлов. Первый содержит все метаданные кроме сообщений, второй - только сообщения.

Для загрузки данных перейдите в блок "Настройка" → "Импорт".

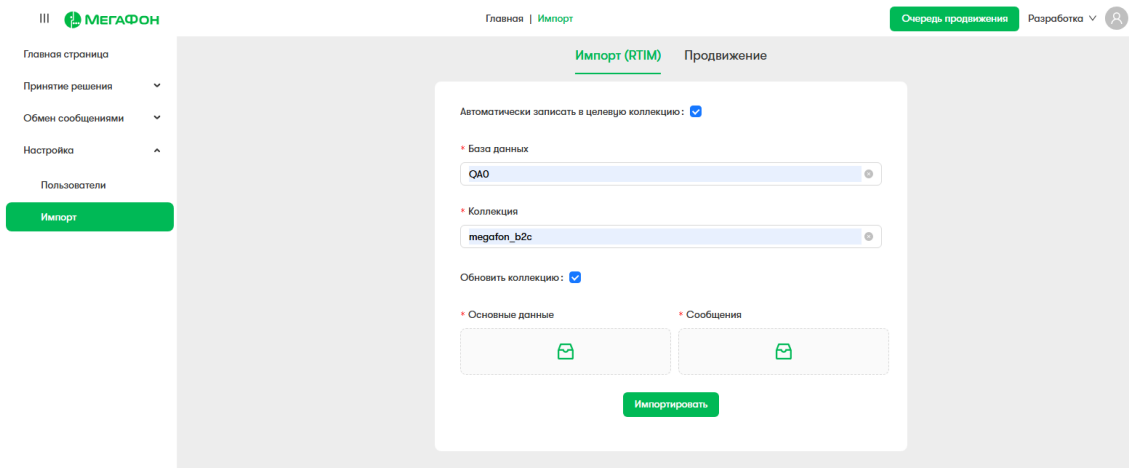


Рисунок 7.2. Импорт данных

На табе "Импорт (RTIM)" отображены:

- Чек-бокс "Автоматически записать в целевую коллекцию" раскрывает дополнительные поля для указания целевой базы и коллекции.
- "База данных" поле для ввода названия базы, в которую будет записан импорт преобразованного бэкапа. Если база данных не существует, она будет создана.
- "Коллекция" поле для ввода названия коллекции, в которую будет записан импорт преобразованного бэкапа. Если коллекция не существует, она будет создана.
- Чек-бокс "Обновить коллекцию" включен по умолчанию. Игнорируется если не выбрана запись в целевую коллекцию. Отвечает за обновление данных - обновит все документы с совпадающим идентификатором, добавит объекты которых не было.

В "Основные данные" необходимо добавить файл с метаданными, в "Сообщения" - файл с сообщениями, соответственно.

Кнопка "Импортировать" запускает процесс загрузки данных в базу и их преобразование. После успешной загрузки в правом верхнем углу будет выведено уведомление "Бэкап успешно импортирован".

Для точечного переноса данных перейдите на таб "Продвижение". На форме представлены:

- "Коллекция" поле с выпадающим списком значений коллекций с импортированными и преобразованными данными "import\_YYYYMMDDTHH24MISS".
- "Целевая среда" поле с выпадающим списком сред: Разработка, Тестирование, Производство.
- "Тип объекта" поле с выпадающим списком значений типов объектов для фильтрации значений в списке с объектами.
- "Объект" поле с выпадающим списком отфильтрованных по типу значений из наименования и ключа объекта.

В зависимости от типа объекта могут быть доступны кнопки: "Продвижение компонента" и/или "Продвижение всех".

Для объектов с типом: Область, Сообщение и Правило доступно только одиночное продвижение (продвижение компонента). Для объектов типа Точка взаимодействия доступно только рекурсивное продвижение (продвижение всех). Для остальных типов Канал и Группа сообщений доступны оба типа продвижения (обе кнопки).

## 8.4 Импорт (RITM)

Система позволяет импортировать данные из экспортированного архива из MegaRITM.

Рисунок 7.3 Импорта данных в MegaRITM

На форме представлены:

- Вид импорта
  - Импорт одной бизнес среды
  - Импорт истории продвижений
  - Импорт конфигурации маппетов
- База данных
- Коллекция
- Чекбокс "Автоматически записать в целевую коллекцию"
- Чекбокс "Обновить коллекцию"
- Поле загрузки файла

Поля "База данных" и "Коллекция" содержат подсказки со списком существующих баз и коллекций, но не ограничивают пользователя. Можно написать любое значение и система запустит импорт и создаст новую коллекцию.

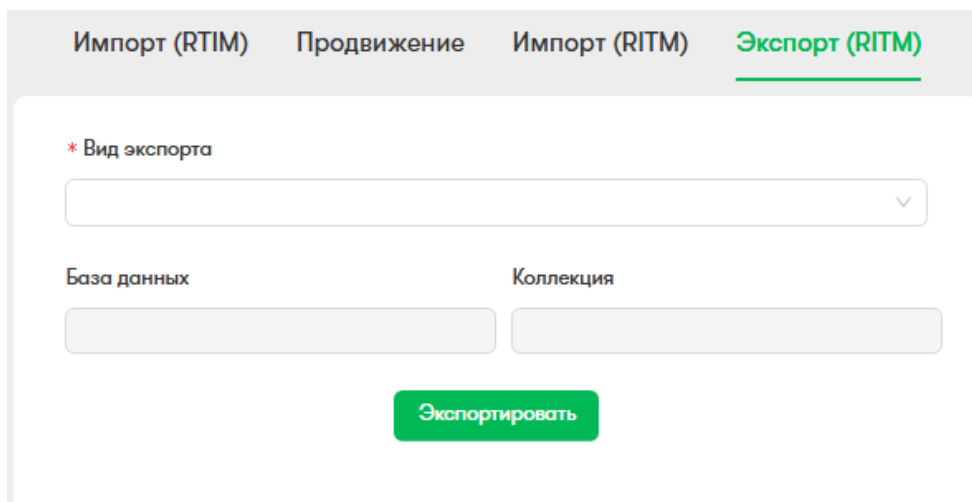
При импорте коллекции она сохраняется в отдельную папку "import", где хранится до 5 последних импортированных файлов для каждого типа (для очереди продвижения, для импорт бизнес-сред, для конфигурации маппетов и тд). При сохранении шестого импорта самый старый будет удален. При наличии чекбокса "Автоматически записать в целевую коллекцию" происходит перезапись данных в целевой коллекции. При этом система ищет объекты с одинаковыми идентификаторами и перезаписывает их новыми значениями, старая версия сохраняется в архивную коллекцию. Новые объекты сохраняются в базу. Бэкап может оказаться некачественным (происходит проверка наличия всех атрибутов в выражениях (в арбитраже и правилах)), в этом случае импорт будет приостановлен.

Чекбокс "Обновить коллекцию" недоступен для технических коллекций.

Если пользователь попытается импортировать файл неподходящего типа (например, техническую коллекцию в бизнес-среду), возникнет ошибка.

## 8.5 Экспорт (RITM)

Система позволяет экспортировать данные из разных бизнес сред и разного типа.



Импорт (RTIM) Продвижение Импорт (RITM) **Экспорт (RITM)**

\* Вид экспорта

База данных Коллекция

Экспортировать

Рисунок 7.4 Экспорт данных из MegaRITM

На форме представлены:

- Вид экспорта
  - Экспорт всех бизнес-сред
  - Экспорт одной из бизнес-сред
  - Экспорт всех технических коллекций
- База данных
- Коллекция

Поля "База данных" и "Коллекция" содержат подсказки со списком существующих баз и коллекций, но не ограничивают пользователя. Можно написать любое значение и система запустит экспорт, если такая коллекция существует (например, архивная коллекция).

Экспорт занимает около 20 секунд, экран при этом неактивен. В результате сохраняется архив с json файлом, содержащим данные как в MongoDB.

При экспорте технических коллекций можно выгрузить историю продвижений (promotion\_queue) или конфигурацию маппетов (conf).

Запрос экспорта защищен аутентификацией либо через логин/пароль в UI, либо через токен в запросе.

## 9 Функции выражения

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
avg	1 если расчет по списку; 2 если обращение к атрибуту комплексного домена	Первый аргумент всегда должен быть списком. Второй строкой.		<p>Возвращает среднее значение для набора значений.</p> <p>В следующем примере находят значения всех атрибутов цен в комплексной категории атрибутов «cart», складывают цены, делят сумму на количество цен и возвращают среднее значение. Оставшаяся часть формулы затем берёт это среднее значение и возвращает значение «истина», если среднее значение больше 1.</p> <pre>avg (   get ("Account.cart"),   "price" ) &gt; 1</pre> <p>По умолчанию функция avg обращается к самым последним данным. Модификатор Session позволяет функции avg обращаться к данным сеанса, то есть ко всем данным активного сеанса. Например, модификатор Session обращается не только к текущему значению атрибута, но и ко всем предыдущим значениям этого атрибута в сеансе. Используйте модификатор Session, чтобы использовать простой атрибут вместо комплексного с функцией avg.</p> <p>Для всех "плоских" категорий атрибутов мы оставили только одно значение "Текущий" в поле "Отбор" при задании критериев для атрибута в конструкторе правил.</p> <p>Session - добавляется как только Отбор != Текущий.</p>
between	2	CustomInt, CustomString, Time		<p>Возвращает все значения между заданными параметрами, включая начальное и конечное значения. Функция between используется только в контексте фильтра или его функций.</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
		Проверка на непустоту		<pre>count(   filter(     get("Account.cart"),     "price",     between(40,45)   ) ) == 3</pre> <p>В примере возвращается значение true, если есть три строки со значением цены, которая больше или равна 40 и меньше или равна 45.</p>
conditional	3			<p>Управляет вычислением других функций в выражении. Условная функция обычно используется в качестве первой функции в стеке выражений.</p> <pre>conditional(   get("Account.dates.expiration"),   NaN,   false ) &amp;&amp; set("fixedDateMillis",   time(get(get("Account.dates.expiration " ),0), "day") ) &amp;&amp; fixedDateMillis &gt; time("2099-01-01T00:00:00.000Z", "day")</pre> <p>Это выражение возвращает значение false, если в сеансе нет истории тегов для дат истечения срока действия. Вычисление выражения останавливается после условной функции, поскольку значение тега равно NULL, а функция get возвращает значение, отличное от числа (NaN), которое является вторым параметром условной функции. Третий параметр в данном случае - это значение всего выражения. Если есть</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
				история тегов для дат истечения срока действия, вычисляется оставшаяся часть выражения.
contains	1	int, CustomString Проверка на непустоту	да	<p>Возвращает все значения, содержащиеся в другом списке значений. Функция contains используется только в контексте фильтра или функций is.</p> <p><b>Пример 1:</b></p> <pre>count(   filter(     get("Account.cart"),     "items",     contains(list("3", "4"))   ) ) &gt; 0</pre> <p><b>Пример возвращает true, если есть строки для элементов со значениями 3 или 4.</b></p> <p><b>Пример 2:</b></p> <pre>count(   filter(     get("Account.cart"),     "products",     contains(get("Account.clearance.products"))   ) ) &gt; 0</pre> <p><b>Пример возвращает значение true, если для товаров в корзине есть строки, входящие в список товаров со скидкой.</b></p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
count	1 Может принимать на вход функции, которые возвращают список			<p>Возвращает количество элементов списка данных в наборе.</p> <p>В следующем примере выражение получает данные сеанса для атрибута Customer.nav.page и фильтрует их по значениям атрибутов, начинающимся со слова Credit. Выражение подсчитывает количество отфильтрованных атрибутов и возвращает значение true, если это число больше или равно 3.</p> <pre>count (   filter(     get("Session.Customer.nav.page"),     pattern("Credit.*")   ) ) &gt;= 3</pre> <p>По умолчанию функция count обращается к самым последним данным. Модификатор Session позволяет функции count обращаться к данным сеанса, то есть ко всем данным активного сеанса. Например, модификатор Session обращается не только к текущему значению атрибута, но и ко всем предыдущим значениям этого атрибута в сеансе. Используйте модификатор Session, чтобы использовать простой атрибут с функцией avg.</p>
distinct	2 Может принимать на вход функции, которые возвращают список			<p>Возвращает список значений, которые являются уникальными в пределах набора.</p> <p>В следующем выражении функция distinct возвращает набор уникальных значений атрибута «price» из категории комплексных атрибутов «cart». Следующее выражение принимает значение «true», если количество уникальных значений атрибута «price» из категории комплексных атрибутов «cart» больше 13.</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
				<pre>count (     distinct (         get ("Account.cart"),         "price"     ) ) &gt; 13</pre> <p>По умолчанию функция distinct обращается к самым последним данным. Модификатор Session позволяет функции distinct обращаться к данным сеанса, то есть ко всем данным активного сеанса. Например, модификатор Session обращается не только к текущему значению атрибута, но и ко всем предыдущим значениям этого атрибута в сеансе. Используйте модификатор Session, чтобы использовать простой атрибут с функцией distinct.</p>
equal_to	1 На вход получает список значений, либо функции, которые возвращают список	CustomInt, CustomString, Time  Проверка на непустоту	да	<p>Возвращает значение true для числового значения атрибута, равного заданному значению. Функция equalTo используется только в контексте фильтра или функций is.</p> <p>В следующем выражении функция equalTo возвращает значение true, когда комплексная категория атрибутов «cart» содержит два товара, у которых значение атрибута «price» равно 1,25.</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
				<pre>count(   filter(     get("Account.cart"),     "price",     equalTo(1.25)   ) ) == 2</pre>
filter	4, 3, 2			<p>Возвращает отфильтрованный список значений путем проверки условия. В следующем примере определяется, есть ли в комплексной категории атрибутов «cart» три атрибута "product" со значением "ToyTruck". Выражение получает все атрибуты комплексной категории атрибутов «cart». Затем выражение отбирает все атрибуты "product" со значением "ToyTruck". Выражение принимает значение true, если в категории «cart» есть три атрибута "product" со значением "ToyTruck".</p> <pre>count(   filter(     get("Account.cart")     "product",     "ToyTruck"   ) ) == 3</pre>
format_date	2			<p>Принимает значение datetime, выраженное как время в миллисекундах с 00:00:00 по Гринвичу 1 января 1970 г., и возвращает его в указанном шаблоне выражения даты Java.</p> <p>Следующее выражение возвращает значение true, если текущая дата — 17 января.</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
				<pre>formatDate(now, "MM-dd") == "01-17"</pre>
format_date_time	1	CustomString Проверка на непустоту	Только в контексте фильтра	<p>Форматирует каждое значение времени в соответствии с указанным временным интервалом. Функция formatDateTime используется только в контексте фильтра.</p> <pre>count(   filter(     get("Account.dates"),     "expiration",     formatDateTime("month"),     equalTo(time(now, "month"))   ) ) &gt; 3</pre> <p>В этом примере возвращается значение true, если в этом месяце имеется более трех дат истечения срока действия.</p>
formatTimePeriod	1		Только в контексте фильтра	<p>Форматирует каждое значение времени в числовое представление указанного временного интервала. Функция formatTimePeriod используется только в контексте фильтра.</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
				<p>Пример 1:</p> <pre data-bbox="1115 400 2045 679">count(   filter(     get("Account.dates"),     "maturity",     formatTimePeriod("day"),     equalTo(15)   ) ) &gt; 3</pre> <p>В этом примере возвращается значение true, если на 15-е число месяца приходится более трех дат погашения.</p> <p>Пример 2:</p> <pre data-bbox="1115 807 2045 1086">count(   filter(     get("Account.dates"),     "maturity",     formatTimePeriod("month"),     equalTo(12)   ) ) &gt; 3</pre> <p>В этом примере возвращается значение true, если в декабре (12-м месяце) имеется более трех дат погашения.</p>
get	1, 2			<p>Возвращает значение одного или нескольких значений атрибутов или списков. Для комплексной категории атрибутов функция get возвращает список записей атрибутов.</p> <p>Чтобы получить значение атрибута необходимо обращаться к нему по полному пути.</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
				<p>Первый уровень:  Customer - служебный объект для плоских доменов,  Session.Account - модификатор для комплексных доменов, когда Отбор != Текущий + служебный объект для комплексных доменов  Messaging - служебный объект для получения параметров сообщения.</p> <p>Второй уровень - наименование категории атрибута.  Третий - наименование атрибута, только для плоских доменов.</p> <p>Например:  Получаем дату рождения абонента из плоского домена subscriber</p> <pre data-bbox="1115 692 2045 783">get (Customer.subscriber.birthDate)</pre> <p>Получение домена "Агрегированная по деревьям история откликов"</p> <pre data-bbox="1115 858 2045 949">get (Session.Account.responsesInpAgr)</pre> <p>Получаем значение атрибута персонализации "Идентификатор родительского предложения" из "Служебной" категории атрибутов персонализации для Сообщения</p> <pre data-bbox="1115 1056 2045 1147">get (Messaging.system.CIM_Collateral_Id)</pre>
greater_than	1	CustomInt, CustomString, Time  Проверка на непустоту	да	Возвращает значение true для числового значения атрибута, превышающего заданное значение. Функция greater_than используется только в контексте фильтра или функций is.

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
				<p>В следующем выражении функция <code>greater_than</code> возвращает значение <code>true</code>, когда в комплексной категории атрибутов «<code>cart</code>» есть два товара, у которых значение атрибута «<code>price</code>» больше 1,25 .</p> <pre>count(   filter(     get("Account.cart"),     "price",     greaterThan(1.25)   ) ) == 2</pre>
<code>greater_than_or_equal</code>	1	CustomInt, Time	да	<p>Возвращает все значения, большие или равные заданным значениям параметра. Функция <code>greater_than_or_equal</code> используется только в контексте фильтра или функций <code>is</code>.</p> <pre>count(   filter(     get("Account.cart"),     "price",     greaterThanOrEqualTo(40)   ) ) == 3</pre> <p>В примере возвращается значение <code>true</code>, если есть три строки со значением атрибута «<code>price</code>», большим или равным 40.</p>
<code>index</code>	2 Первый аргумент всегда должен быть списком,	CustomInt, CustomString Проверка на непустоту		<p>Возвращает индекс записи, соответствующей внутренней функции фильтра.</p> <p>Следующее выражение всегда возвращает значение <code>true</code>.</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
	второй - функция либо значение, по которому надо искать в первом списке			<pre>index(   list("a", "b", "c"),   "b" ) == 1</pre> <p>Это индекс, начинающийся с нуля, поэтому при совпадении с первой записью возвращается индекс 0. Функция индекса возвращает -1, если указанное значение не найдено.</p>
is	2	Проверка на непустоту		<p>Проверяет, соответствует ли заданное значение другому.</p> <pre>is(   get("Customer.demographics.age"),   between(25,50) )</pre> <p>Это выражение принимает значение true, если возраст клиента составляет от 25 до 50 лет.</p>
less_than	1	CustomInt, CustomString, Time	да	<p>Возвращает значение true, если числовое значение атрибута меньше заданного. Функция less_than используется только в контексте фильтра или функций is.</p> <p>В следующем выражении критерия функция less_than возвращает значение true, когда комплексная категория атрибутов «cart» содержит два товара, у которых значение атрибута «price» меньше 1,25.</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
				<pre>count(   filter(     get("Account.cart"),     "price",     lessThan(1.25)   ) ) == 2</pre>
less_than_or_equal	1	CustomInt, CustomString, Time	да	<p>Возвращает все значения, которые меньше или равны заданным значениям параметра. Функция less_than_or_equal используется только в контексте фильтра или функций is.</p> <pre>count(   filter(     get("Account.cart"),     "price",     lessThanOrEqual(40)   ) ) == 3</pre> <p>В примере возвращается значение true, если есть три строки со значением атрибута «price», меньшим или равным 40.</p>
list	1 - ∞			<p>Возвращает временный список элементов.</p> <p>Следующее выражение критерия создает список: a, b, c и всегда возвращает true, поскольку элемент, расположенный вторым в списке (обозначенный 1), - это b.</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
				<pre>get (   list ("a", "b", "c"),   1 ) == "b"</pre>
max	1, 2			<p>Возвращает максимальное значение из списка значений.</p> <p>В следующем примере выражение возвращает значение true, если максимальное значение для любого атрибута «price» в комплексной категории атрибутов «cart» больше 100.</p> <pre>max (   get ("Account.cart"),   "price" ) &gt; 100</pre> <p>Если записей для атрибута «price» нет, функция max возвращает 0.</p> <p>По умолчанию функция max обращается к самым последним данным. Модификатор Session позволяет функции max обращаться к данным сеанса, то есть ко всем данным активного сеанса. Например, модификатор Session обращается не только к текущему значению атрибута, но и ко всем предыдущим значениям этого атрибута в сеансе. Используйте модификатор Session, чтобы использовать простой атрибут с функцией max .</p> <p>Функцию max можно использовать с датами.</p>
min	2			<p>Возвращает минимальное значение из списка значений.</p> <p>В следующем примере выражение критерия возвращает значение true, если минимальная цена в комплексной категории атрибутов «cart» больше 12.</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
				<pre data-bbox="1122 359 2045 523"> min(   get("Account.cart"),   "price" ) &gt; 12 </pre> <p data-bbox="1113 528 2045 560">Если записей для атрибута «price» нет, функция min возвращает 0 .</p> <p data-bbox="1113 571 2045 786">По умолчанию функция min обращается к самым последним данным. Модификатор Session позволяет функции min обращаться к данным сеанса, то есть ко всем данным активного сеанса. Например, модификатор Session обращается не только к текущему значению атрибута, но и ко всем предыдущим значениям этого атрибута в сеансе. Используйте модификатор Session , чтобы использовать простой атрибут с функцией min.</p> <p data-bbox="1113 802 2045 834">Функцию min можно использовать с датами.</p>
nan				<p data-bbox="1113 863 2045 919">Возвращает true, если значение атрибута, которое должно было быть математическим числом, не является числом.</p> <p data-bbox="1113 935 2045 1023">Следующее выражение всегда возвращает значение true, поскольку недопустимое среднее значение (среднее значение ничего) не является числом.</p> <pre data-bbox="1122 1043 2045 1129"> avg("") == NaN </pre> <p data-bbox="1113 1134 2045 1198">Использование функции значения в текстовой строке приводит к ошибке, а не к NaN (не числу).</p>
notEqualTo		CustomInt, float, CustomString, Time	да	<p data-bbox="1113 1227 2045 1315">Возвращает все значения, не равные заданным значениям параметра. Функция notEqualTo используется только в контексте фильтра или функций is.</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
		Проверка на непустоту		<pre>count(   filter(     get("Account.cart"),     "price",     notEqualTo(40)   ) ) == 3</pre> <p>В примере возвращается значение true, если есть три строки со значением атрибута «price», не равным 40.</p>
now				<p>Возвращает текущую дату и время (выраженную в виде времени в миллисекундах с 00:00:00 по Гринвичу 1 января 1970 г.), которое, в свою очередь, может быть передано функции времени для использования в математических операциях с датами.</p> <p>В следующем выражении функция now используется с форматированием даты. Выражение возвращает значение true, если текущая дата — 17 января.</p> <pre>formatDate(now, "MM-dd") == "01-17"</pre>
parse_date		CustomInt, CustomString,		<p>Принимает отформатированную строку даты и преобразует ее в миллисекунды с 00:00:00 по Гринвичу 1 января 1970 года.</p> <pre>parseDate(get("Customer.demographics.dob"))</pre> <p>В следующем примере проверяется, находится ли дата рождения клиента менее чем в 30 днях от текущего времени:</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
				<pre>time(now) - time(parseDate(get("Customer.demographics.dob"))) &lt; 86400000 * 30</pre> <p>В предыдущей формуле используется число 86400000, поскольку это количество миллисекунд в сутках. В качестве альтернативы вы можете включить математические выражения для вычисления количества миллисекунд в сутках, как показано в следующем примере:</p> <pre>time(now) - time(parseDate(get("Customer.demographics.dob"))) &lt; 1000 * 60 * 60 * 24 * 30</pre>
pattern		CustomString Проверка на непустоту		<p>Создает регулярное выражение.</p> <p>Функция pattern использует класс Java Pattern для определения регулярного выражения. Функция pattern может включать тестовое значение для оценки или использоваться совместно с функциями filter и index. Необязательный второй параметр определяет флаги, определяемые классом Java Pattern. Необязательный третий параметр определяет тестовое значение.</p> <p>Следующее выражение всегда возвращает значение true. (Второй параметр в следующем примере - число 2 - это флаг для класса Java Pattern, который указывает на отсутствие чувствительности к регистру. Не путайте этот флаг с использованием числа 2 в большинстве других функций выражения критерия, которое указывает на третий элемент.)</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
				<pre data-bbox="1149 379 1682 403">pattern("Chris .*", 2, "chris Smith")</pre> <p data-bbox="1111 448 1984 531">Ниже приведены примеры регулярных выражений, которые можно использовать с функцией pattern. Значение второго параметра имеет значение, но не является обязательным.</p> <pre data-bbox="1149 579 1973 767">pattern("Candy.*", 2, "CandyBar") // начинается с pattern("^(!Bar).*", 2, " CandyBar ") // не начинается с pattern(".*Bar", 2, " CandyBar ") // заканчивается на pattern("^(!.*Candy\$).*", 2, " CandyBar ") // не заканчивается на pattern(".*Candy.*", 2, " CandyBar ") // содержит pattern("^((?!cb).)*", 2, " CandyBar ") // не содержит</pre> <p data-bbox="1111 810 1984 893">Следующее выражение критерия возвращает true, если комплексная категория атрибутов «cart» включает атрибут «product» со значением, которое заканчивается на Truck.</p> <pre data-bbox="1149 946 1626 1129">count(   filter(     get("Account.cart"),     "product",     pattern(".*Truck")   ) ) != 0</pre>
rand				<p data-bbox="1111 1182 1485 1206">Генерирует случайные числа.</p> <p data-bbox="1111 1230 1962 1313">В следующем примере создается временная переменная r, которой присваивается случайное значение. Выражение вернет true если r окажется больше 0,05.</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
				<pre>set("r", rand()) &amp;&amp; get("r") &gt; 0.05</pre>
range	3			<p>Возвращает true, если среднее значение больше или равно первому параметру и меньше или равно третьему параметру.</p> <p>Ниже показан синтаксис для range:</p> <pre>range (lower_value, middle_value, upper_value)</pre> <p>Например, следующее выражение критерия возвращает значение true, если атрибут Customer.nav.page со значением, имеющим шаблон .Credit.*, присутствует в последних четырех экземплярах этого атрибута.</p> <pre>range (   0,   index (     get ("Session.Customer.nav.page"),     pattern ("Credit.*")   ),   3 )</pre>
select	2			<p>Получает значение одного атрибута. Если атрибут относится к комплексной категории, функция select может получить все значения этого атрибута.</p> <p>Следующее выражение критерия возвращает true, если комплексная категория атрибутов «cart» содержит три атрибута «product».</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
				<pre>count(select("product", get("Account.cart"))) == 3</pre>
set	2			<p>Создает временную переменную, которую можно использовать позже в текущем выражении критерия.</p> <p>В следующем примере переменной X присваивается значение B, поэтому выражение всегда возвращает значение true.</p> <pre>set("X", "B") &amp;&amp; X == "B"</pre>
sum	2			<p>Возвращает сумму набора значений.</p> <p>В следующем примере возвращается значение true, если сумма атрибутов «price» в комплексной категории «cart» больше 1.</p> <pre>sum(   get("Account.cart"),   "price" ) &gt; 1</pre> <p>По умолчанию функция sum обращается к самым последним данным. Модификатор Session позволяет функции sum обращаться к данным сеанса, то есть ко всем данным активного сеанса. Например, модификатор Session обращается не только к текущему значению атрибута, но и ко всем предыдущим значениям этого атрибута в сеансе. Используйте модификатор Session, чтобы использовать простой атрибут с функцией sum.</p>
time	1, 2, 4			<p>Возвращает время (в виде количества миллисекунд от начала эпохи) из поля datetime. Функция time также используется для преобразования заданного поля datetime в указанный временной интервал. Функция</p>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
				<p>поддерживает относительные даты. Функция time работает в часовом поясе UTC.</p> <p>Функция времени поддерживает следующие периоды времени:</p> <ul style="list-style-type: none"> <li>• hour: обрезает заданное время до часов. Например, если time(now,"hour") и сейчас 24.08.2025 13:30, то это преобразуется в сегодняшнюю дату 24.08.2025 13:00.</li> <li>• day: преобразует указанное время в день месяца. Например, для time(now,"day") и сейчас 24.08.2025 13:30, это превращается в сегодняшнюю дату 24.08.2025 00:00.</li> <li>• week: преобразует указанное время в неделю года. Например, time(now,"week") разрешается в datetime, представляющую неделю года в нулевой час (12:00).</li> <li>• month: преобразует указанное время в месяц года. Например, time(now,"month") разрешается в datetime, представляющую месяц года в нулевой час (12:00).</li> <li>• quarter: преобразует указанное время в месяц года, представляющий начало квартала. Например, для time(now,"quarter") , если сейчас 31.12.2024, это преобразуется в 01.10.2024 в нулевой час (00:00).</li> <li>• year: преобразует указанное время в календарный год. Например, для time(now,"year") , если сейчас 31.12.2024, это преобразуется в 01.01.2024 в нулевой час (00:00).</li> </ul>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filter или функции is	Описание
				<p>Пример 1: Как показано в следующем примере, вы можете передать результат <code>parseDate</code> функции <code>time</code>.</p> <pre data-bbox="1115 459 2036 657"> now - time(   parseDate(     get("Customer.demographics.dob")   ) ) &lt; 1000 * 60 * 60 * 24 * 30 </pre> <p>Пример 2: Следующее выражение оценивается как истинное, если сегодня май 2015 года, независимо от дня и времени.</p> <pre data-bbox="1115 767 2036 884"> time(now,"month") == time("2015-05-11T08:20:00.000Z","month") </pre> <p>Пример 3: Функция <code>time</code> также поддерживает относительные даты и периоды времени. Например, <code>time(now,5,"day","past")</code> вычисляет время через 5 дней после заданного времени. Если сейчас 24.08.2025, возвращается 19.08.2025 в нулевой час (00:00). Другой пример: <code>time(now,12,"hour","future")</code> вычисляет время через 12 часов после заданного времени. Если сейчас 9:00, возвращается 21:00.</p> <p>Следующее выражение оценивается как истинное, если дата истечения срока действия наступает ранее, чем через 3 месяца.</p> <pre data-bbox="1115 1195 2036 1310"> time(get("Account.dates.expiration ")) &lt; time(now,3,"month","future") </pre>

Функция	Количество аргументов	Ожидаемые типы данных	Используется только в filters или функции is	Описание
Value	1	CustomInt, CustomString Проверка на непустоту		<p>Преобразует строковое значение в число.</p> <p>Когда число передается как строка, функция value преобразует его в число.</p> <p>Следующее выражение возвращает значение true, если собаке 10 лет (то есть атрибут Customer.dog.age имеет значение 10), поскольку 7, умноженное на 10, равно 70.</p> <pre>7 * value(get("Customer.dog.age")) == 70</pre>