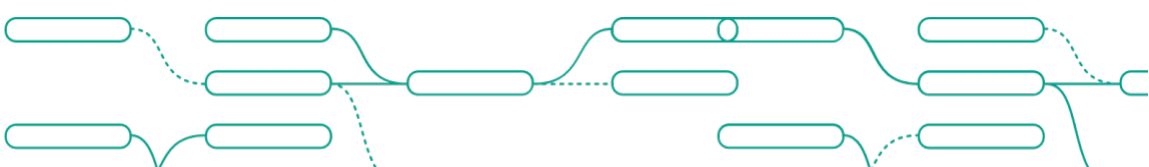
Arenadata Catalog

Arenadata Catalog. AI Стюард данных

Функциональные возможности ADC.AI DS 1.2

Москва 2026



Arenadata Catalog

Оглавление

Журнал изменений **3**
Введение **4**
Термины и определения 4
Сокращения и обозначения 4
Общие положения 4
AI Стюард данных **5**
Агент обогащения метаданных объекта 5
Агент связей 5
Агент генерации происхождения хранимых процедур 6
Агент генерации происхождения из S3 6
Агент генерации происхождения из Git 7
Ручной и массовый запуск задач агентов 8
Отслеживание истории выполнения задач агентами 9
Заявки на изменение от агентов 11

Журнал изменений

Дата	Версия документа	Комментарий
27.07.2026	0.1	Начальная версия документа

Введение

Термины и определения

Термин	Значение
Запуск	Группа задач одного агента, выполняемых с общей конфигурацией и приоритетом
Задача	Единица работы агента над одним объектом (таблицей, термином, процедурой, репозиторием)

Сокращения и обозначения

Сокращение	Наименование
ADC	Arenadata Catalog
AI DS	AI Стюард данных

Общие положения

Настоящий документ является кратким описанием функциональных возможностей модуля AI Стюард данных (AI DS) программного обеспечения Arenadata Catalog (ADC).

ADC – это масштабируемая отказоустойчивая система, предназначенная для удобного и простого формирования базы знаний о ландшафте данных и управления ею. Система реализована на базе OpenMetadata (open source). Значимыми преимуществами Arenadata Catalog являются расширенный функционал Бизнес-глоссария, а также интеграция с другими компонентами платформы Arenadata ED.

AI Стюард данных

AI Стюард данных доступен в виде отдельного модуля системы Arenadata Catalog. Модуль предоставляет функциональные возможности для реализации сценариев применения ИИ-агентов.

Благодаря использованию модуля AI Стюард данных пользователи могут достичь существенного снижения трудозатрат на ведение каталога, улучшения качества и актуальности метаданных.

Предложения агентов обогащения метаданных и связей глоссария могут быть оформлены в виде заявки на изменение для ручного согласования ответственным пользователем либо записаны сразу в объект. Агенты происхождения данных публикуют найденные связи непосредственно в граф происхождения.

Агент обогащения метаданных объекта

Агент актуализирует метаданные объектов данных: предлагает пользовательские описания, теги классификаций и объекты глоссария для связи.

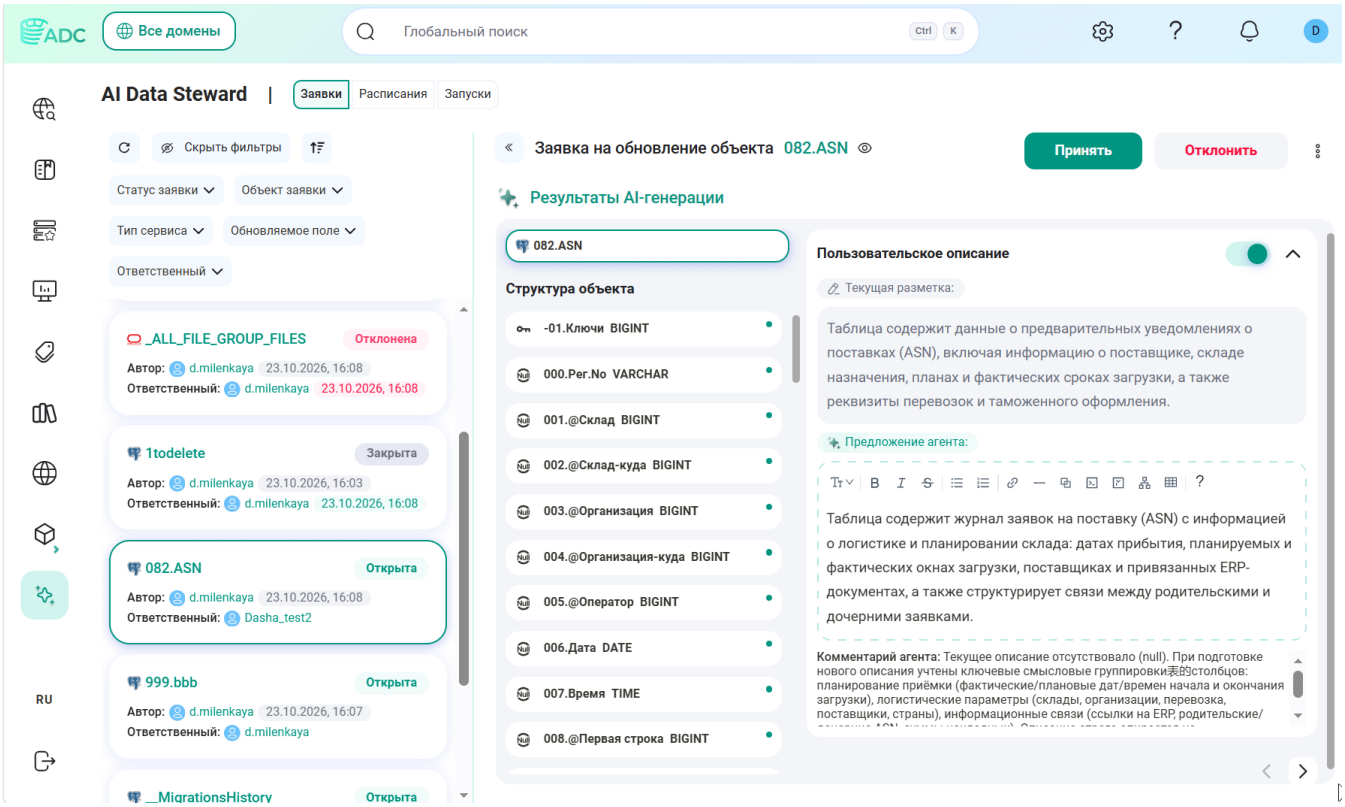


Рисунок 1 – Просмотр открытой заявки из карточки объекта с предложенным пользовательским описанием

Агент связей

Агент подбирает связанные термины глоссария и предлагает установить между ними системную связь «Связанные объекты».

Агент генерации происхождения хранимых процедур

Агент извлекает SQL-код из хранимой процедуры, загруженной в ADC, анализирует его с помощью LLM и строит граф происхождения данных между таблицами.

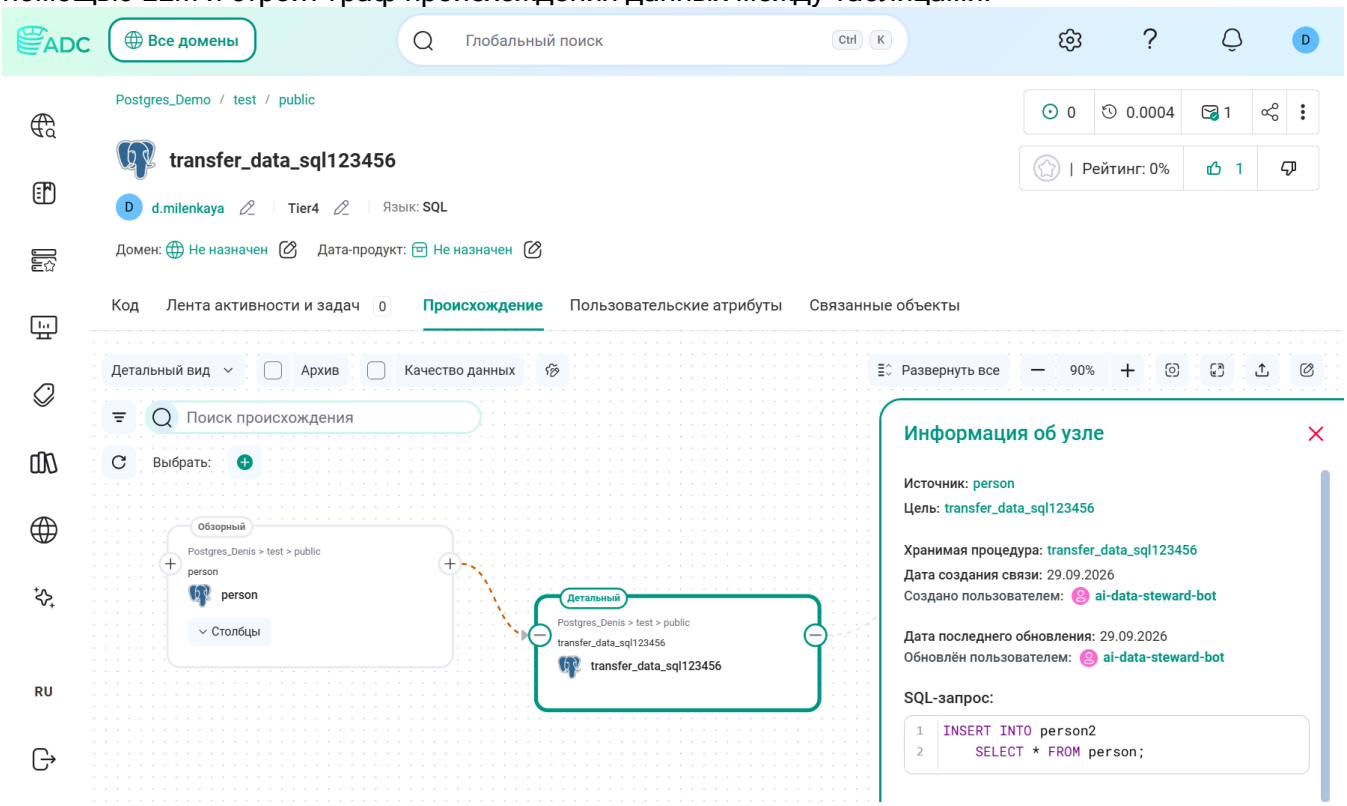


Рисунок 2 – Объект со сгенерированным происхождением

Агент генерации происхождения из S3

Агент извлекает SQL-код из объектов в S3, анализирует его с помощью LLM и строит граф происхождения данных между таблицами

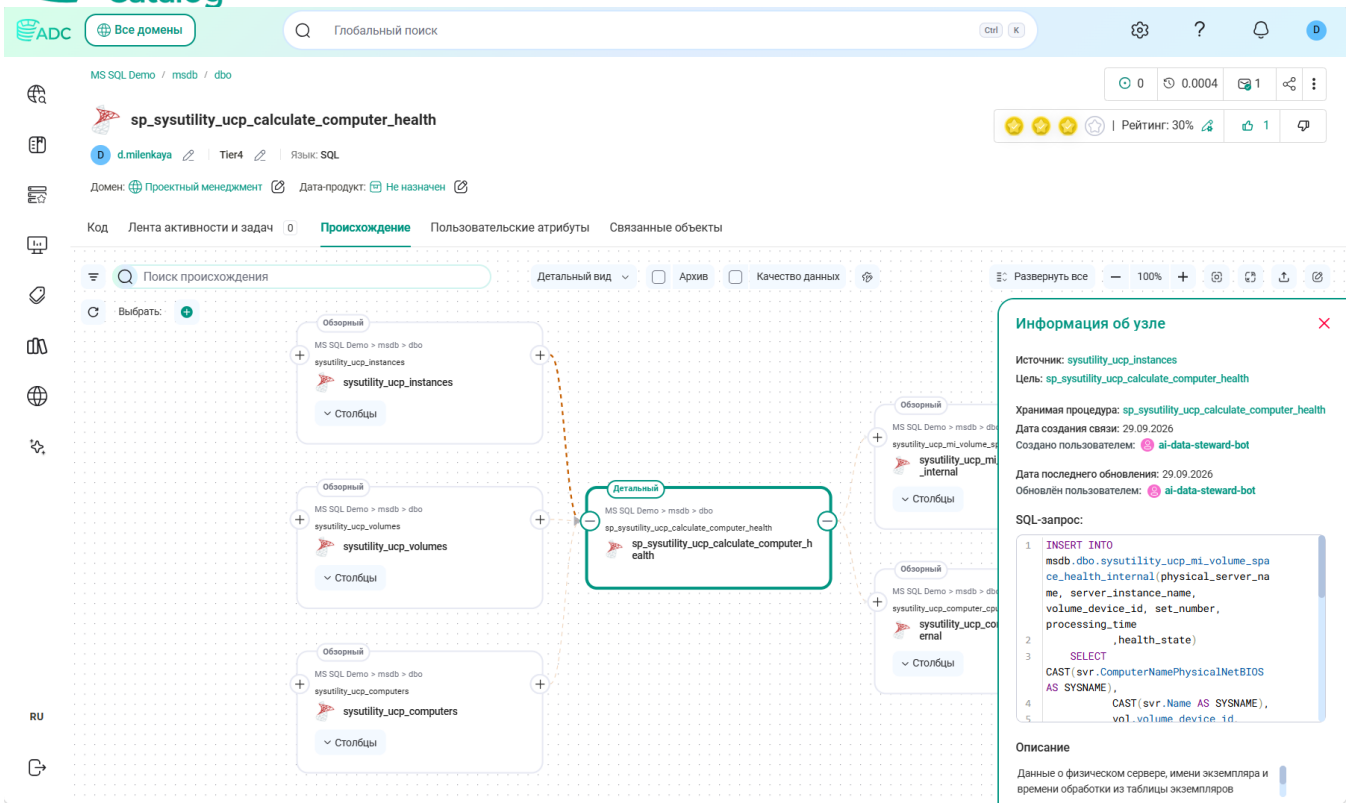


Рисунок 3 – Объект со сгенерированным происхождением

Агент генерации происхождения из Git

Агент извлекает SQL-код из Git-репозитория, анализирует его с помощью LLM и строит граф происхождения данных между таблицами

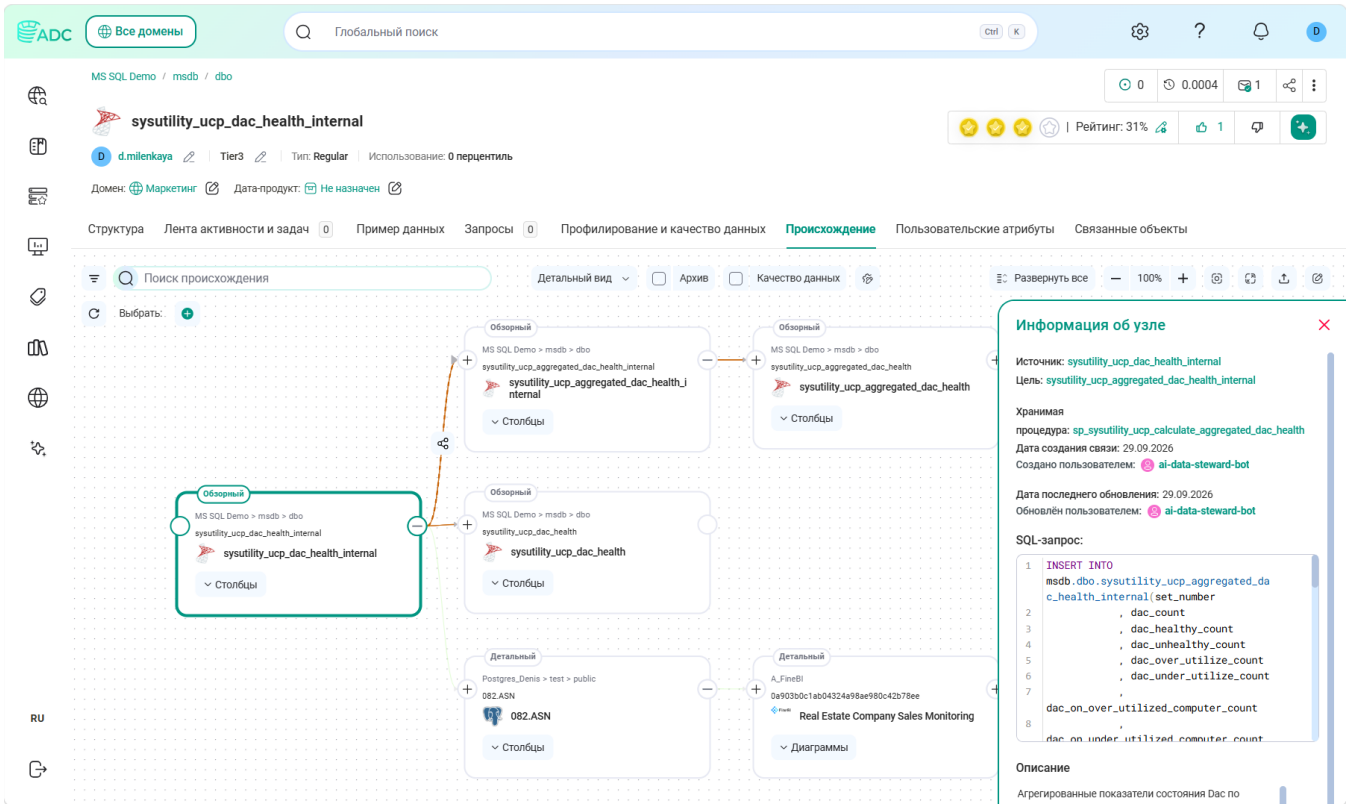


Рисунок 4 – Объект со сгенерированным происхождением

Ручной и массовый запуск задач агентов

Доступен массовый запуск задач по созданному расписанию в интерфейсе системы. В расписании настраиваются параметры:

- Тип задачи генерации;
- Необходимость оформления предложений агентов в виде заявки на изменение;
- Выбор типов объектов;
- Фильтрация сервисов объектов;
- Необходимость актуализации уже существующих значений атрибутов объектов;
- Расписание массового запуска задач.

В случае выбора агента типа «Обогащение метаданных объекта» доступно выбрать, что именно необходимо сгенерировать: пользовательское описание, разметку классификаторами или объектами глоссария – с дополнительным уточнением по доступным категориям классификаторов, глоссариям и типам объектов глоссария.

Для агента происхождения из Git в конфигурации запуска задаются владелец и имя репозитория, ветка и фильтр путей. Для агента происхождения из S3 задаются имя бакета и фильтр ключей объектов.

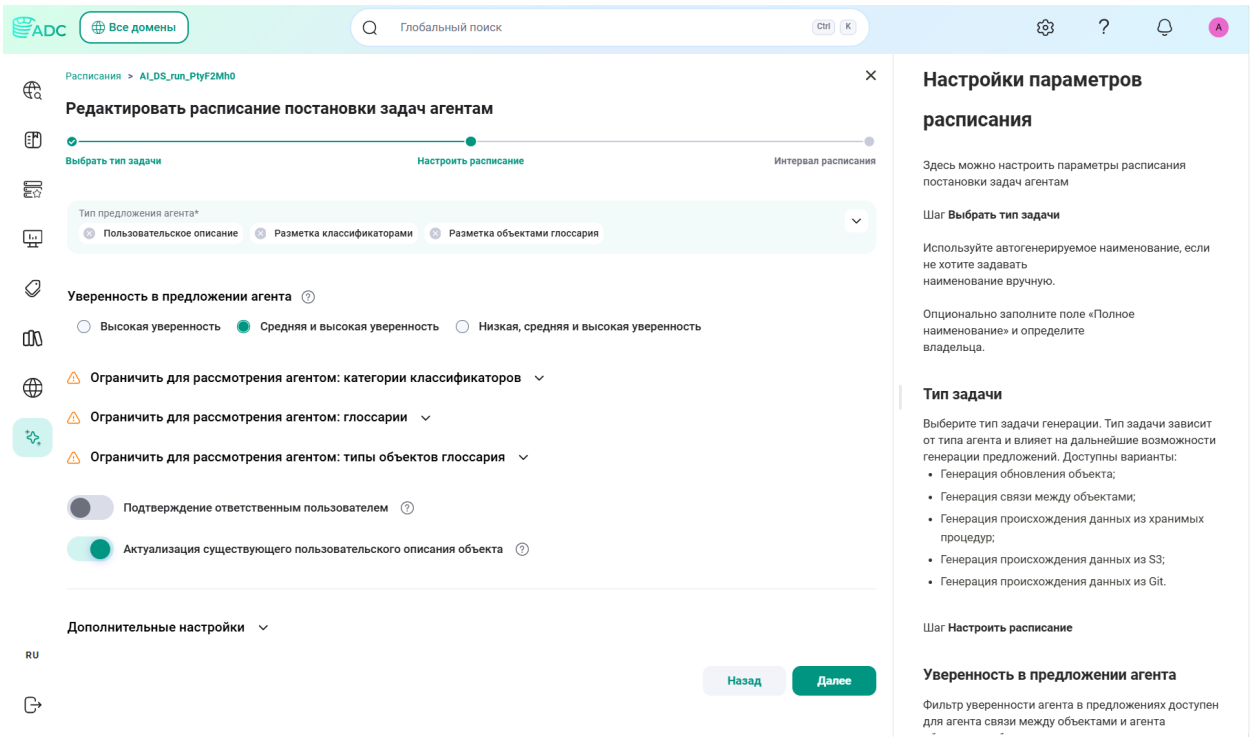


Рисунок 5 – Создание расписания

Ручной запуск возможен из карточки выбранного объекта. В настройках запуска выбирается:

- Необходимость оформления предложений агентов в виде заявки на изменение;
- Для агента обогащения метаданных объекта – выбор категорий классификаторов, глоссария и типов объектов глоссария.

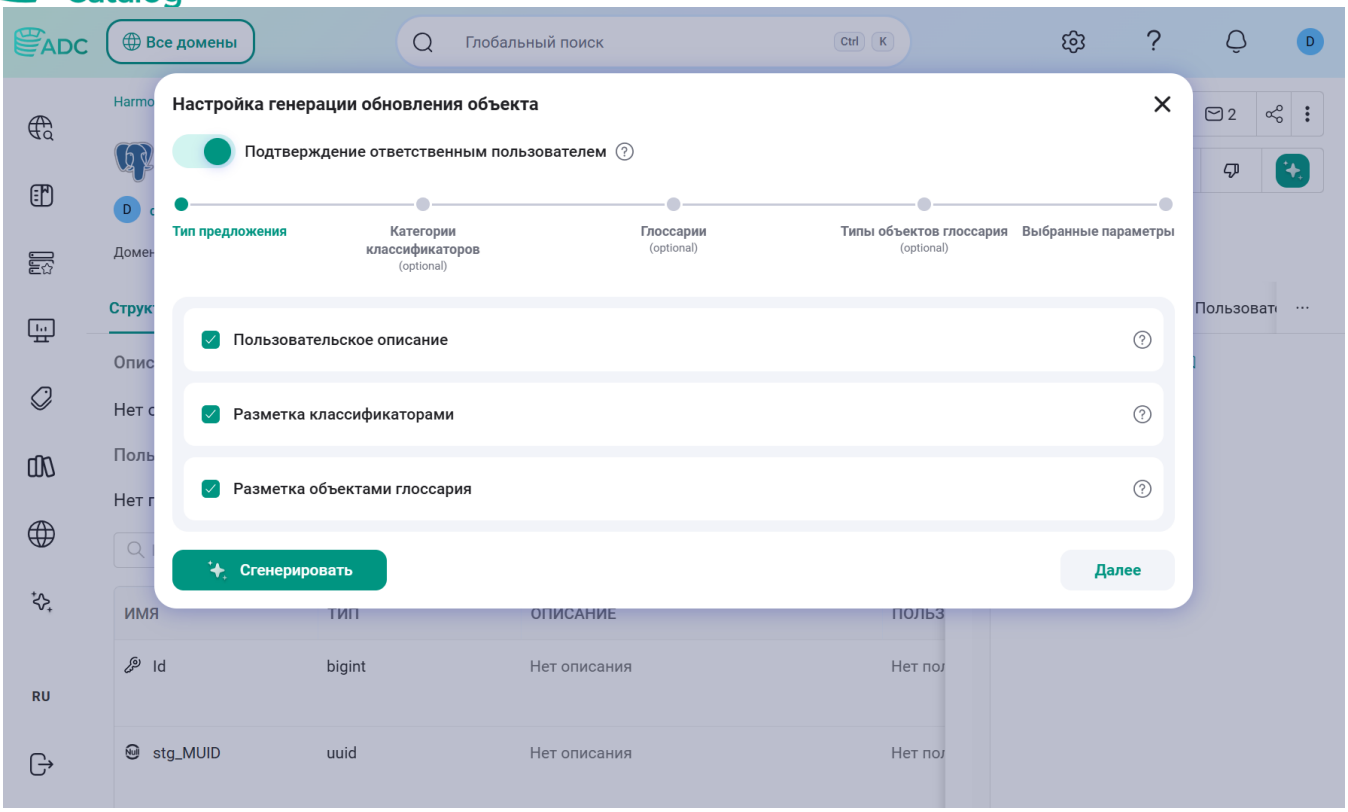


Рисунок 6 – Модальное окно настройки ручного запуска

Отслеживание истории выполнения задач агентами

В системе доступно отслеживание истории выполнения задач агентами для проведения мониторинга в нескольких разрезах.

В карточке объекта заявки возможно просмотреть ранее выполняемые задачи агентов с дополнительной информацией о шагах исполнения задач и результате их выполнения.

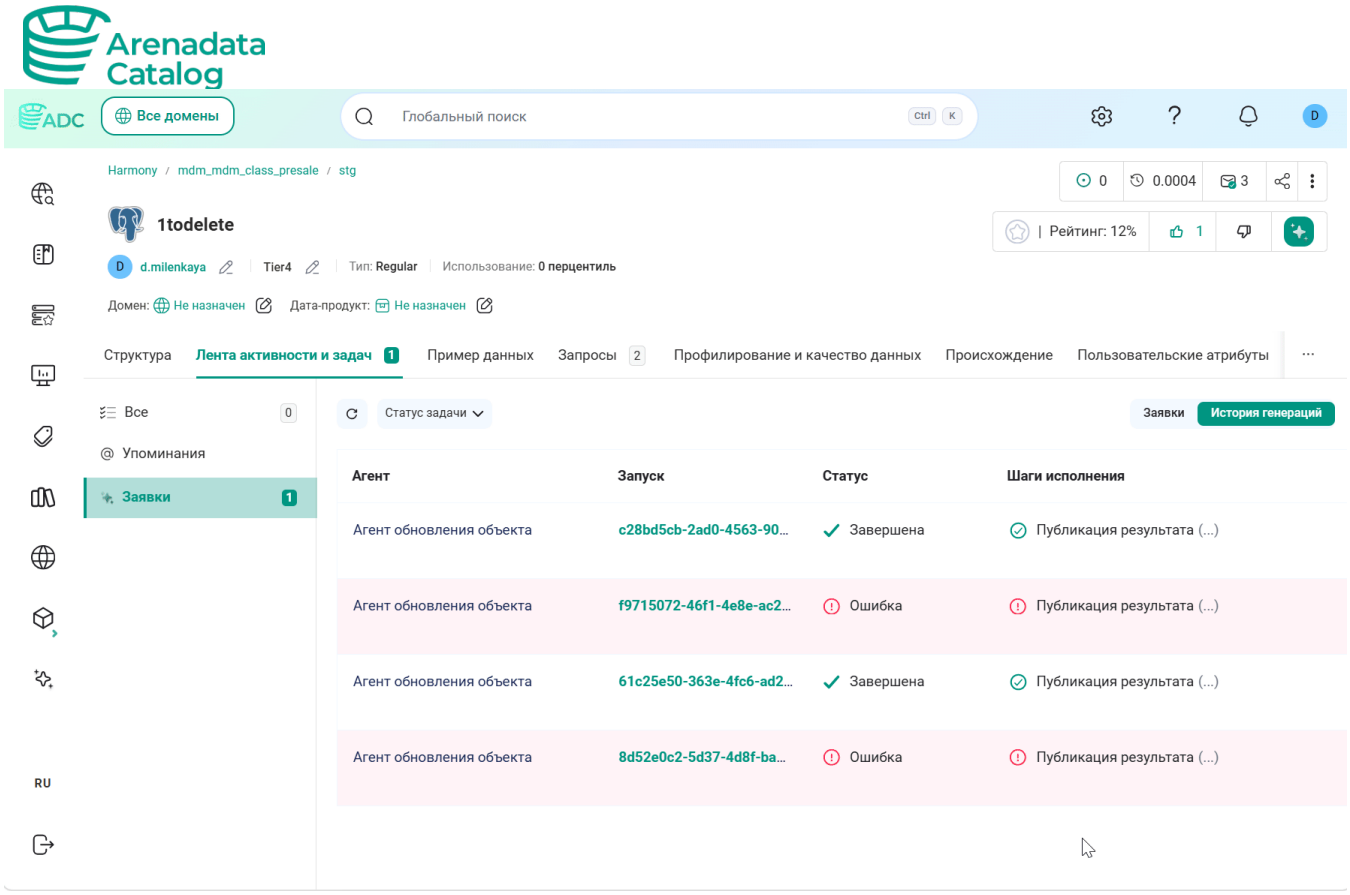


Рисунок 7 – История генерации

В разделе «AI Стюард данных» на вкладке «Расписания» представлена информация о существующих в системе расписаниях запуска. Дополнительно доступно просмотреть историю запусков по каждому расписанию.

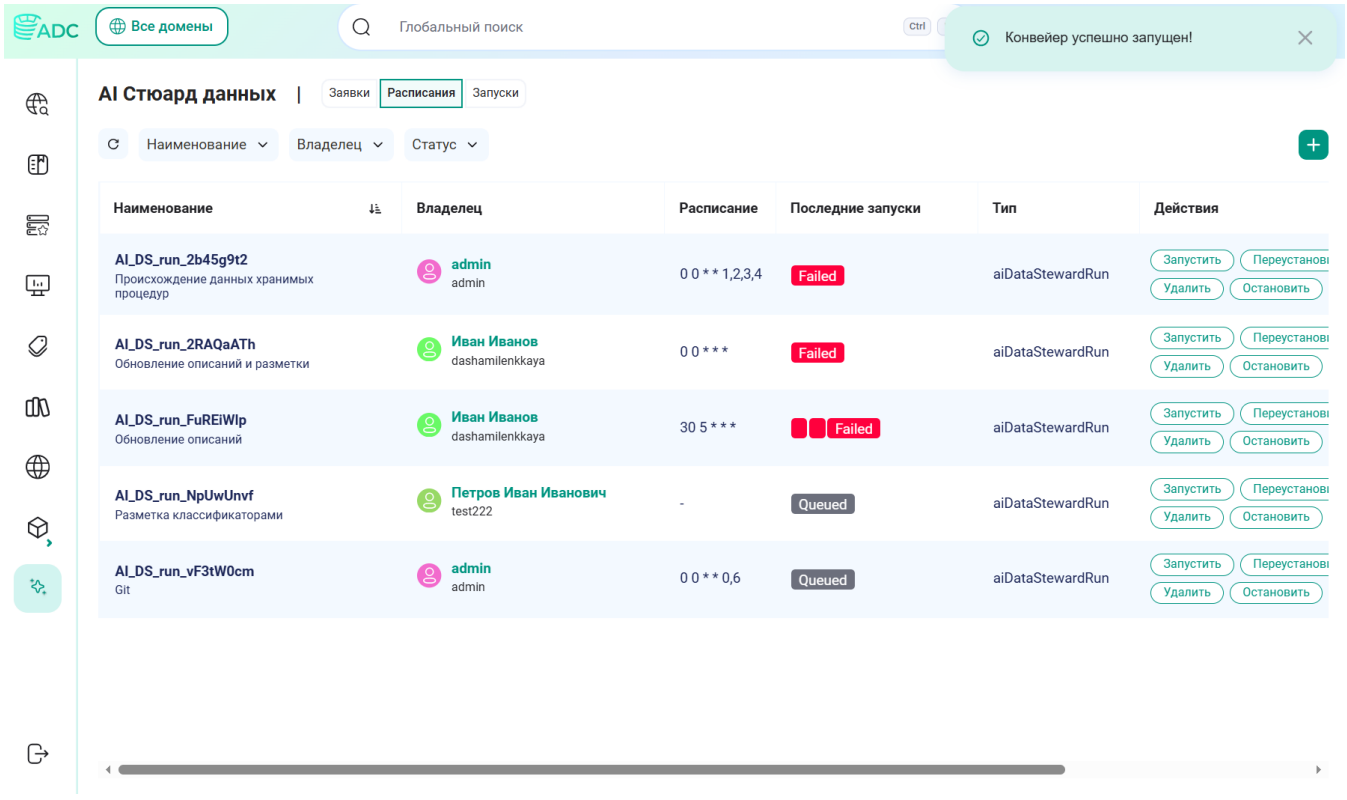


Рисунок 8 – раздел «AI Стюард данных вкладка Расписания»

В разделе «AI Стюард данных» на вкладке «Запуски» представлена информация о всех задачах агентов с группировкой по запускам.

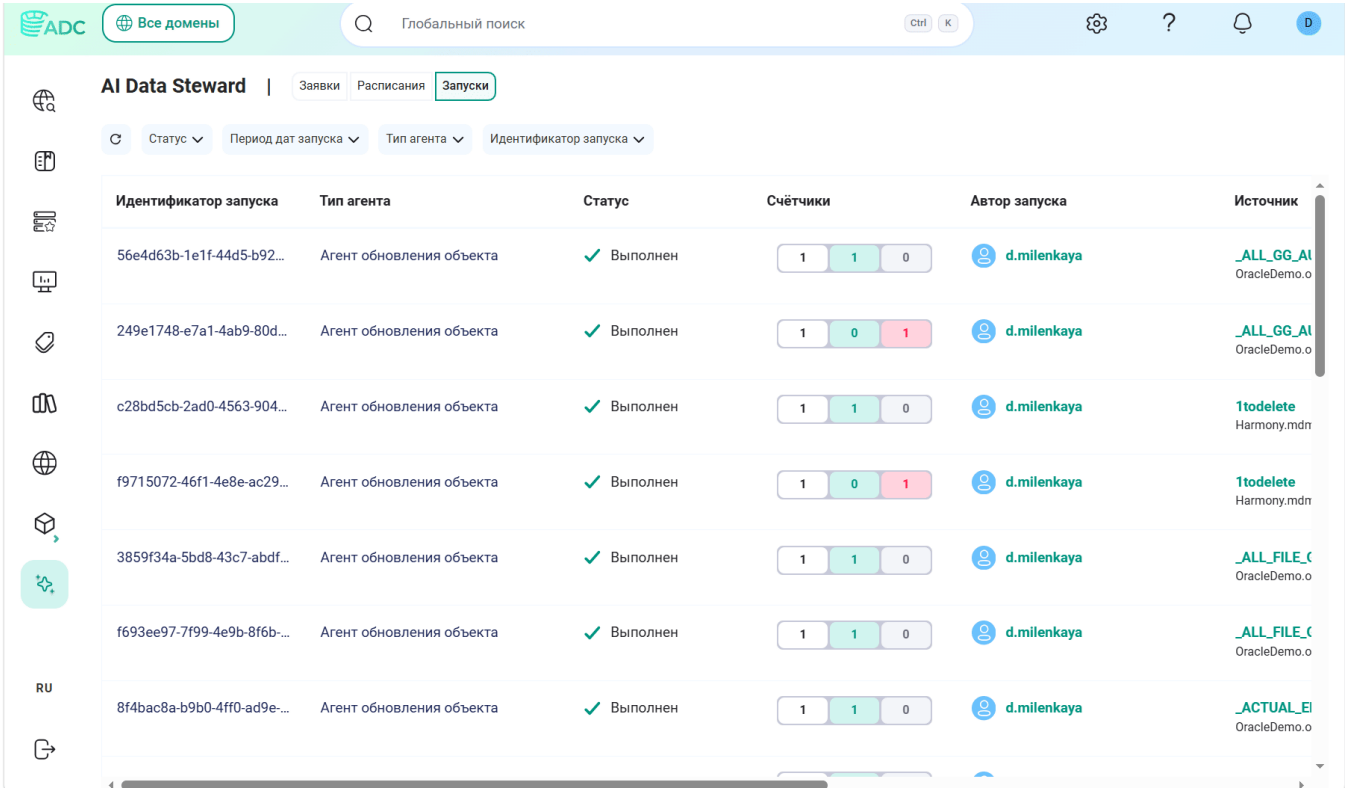


Рисунок 9 – раздел «AI Стюард данных вкладка Запуски»

Заявки на изменение от агентов

Предложения агентов обогащения метаданных и связей глоссария могут быть оформлены в виде заявок на изменение для вынесения решения по ним ответственным пользователем.

Ответственным является владелец объекта или, если таковой не назначен, дефолтный ответственный, определённый в конфигурации агента.

С заявками доступно работать из:

- карточки объекта заявки;
- профиля пользователя;
- раздела «AI Стюард данных».

В зависимости от того, откуда была открыта заявка, её наполнение может незначительно различаться. Например, в заявке из раздела «AI Стюард данных» будет доступен дополнительный блок с информацией об объекте заявки для наглядности текущего состояния объекта в системе.

В заявке также представлен блок с актуальной информацией о значениях атрибута, который агент предлагает обновить.

Ответственный пользователь может выборочно корректировать предложения агента или отклонять их полностью, если какое-либо из них признано ошибочным для объекта внутри заявки.

Если заявка принята, предложения агента будут записаны в объект с изменением его версии. Ранее отклонённые заявки можно переоткрыть, чтобы повторно принять по ним решение.

В закрытых заявках можно получить информацию о предложении, скорректированном пользователем, если были внесены изменения на этапе вынесения решения.

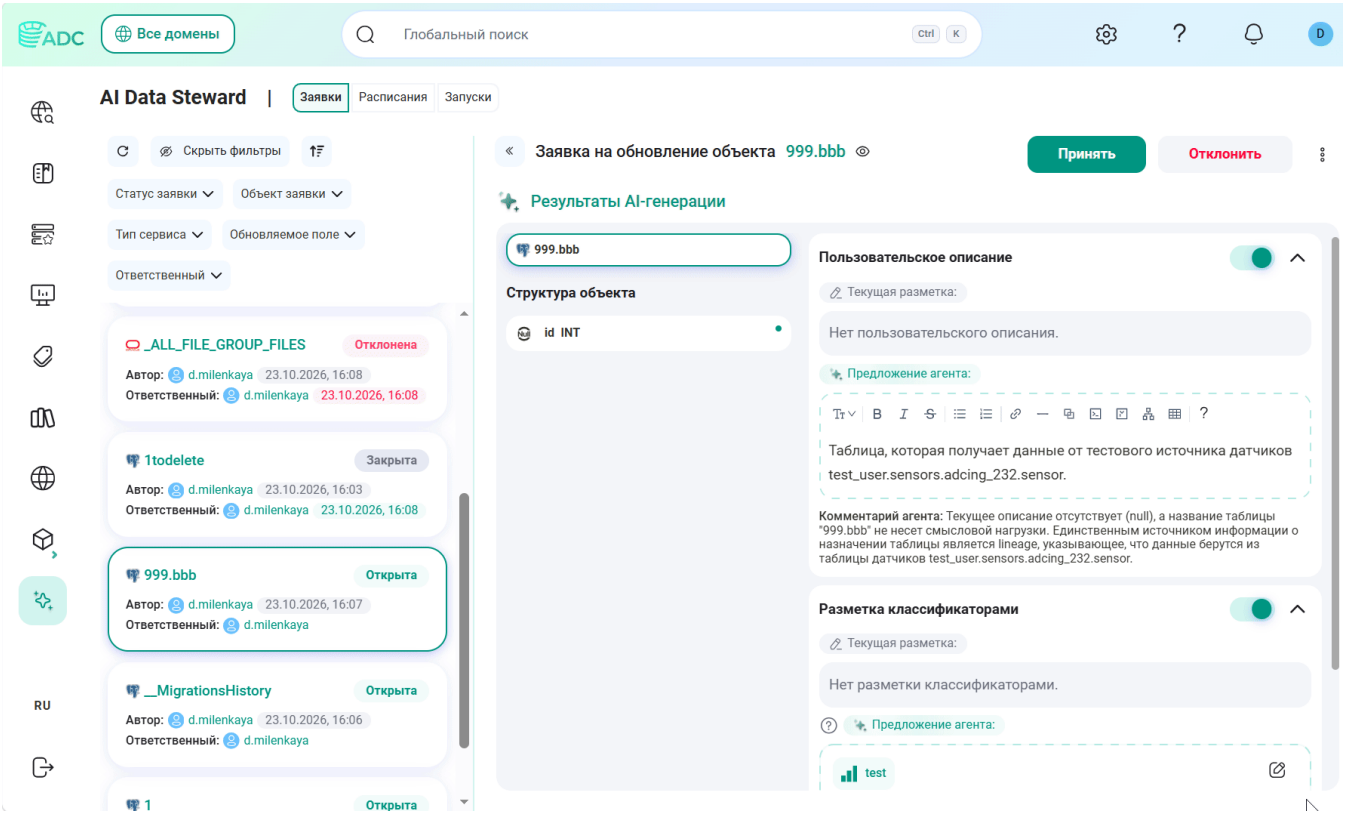


Рисунок 10 – Заявка на изменение от агента