

Руководство по эксплуатации

AI Agents Hub | 0.1.0 | 28.09.2026

Редакция документа 1.5 от 28 сентября 2026 года.

Содержание: 1 Общие сведения · 2 Термины и подготовка пользователя · 3 Роли права и согласования · 4 Запуск и демонстрационные сценарии · 5 HTTP API · 6 Знания и изменение поведения · 7 Аудит и ошибки · 8 Резервное копирование и восстановление · 9 Завершение работы и поддержка

1 Общие сведения

Базовый дистрибутив содержит демонстрационные сценарии управления арендной недвижимостью. Команды с demo-room иллюстрируют проверку установки и работу API; они не ограничивают назначение платформы этой отраслью. Адаптация под предметную область заказчика, включая проектную настройку или дообучение LLM, выполняется отдельно и сопровождается документацией соответствующей конфигурации.

AI Agents Hub, версия 0.1.0. Руководство описывает работу системного администратора и интегратора через командную строку и HTTP API.

Для начала работы выполните установку и активацию профиля по инструкции. Коммерческие условия и обязательства сопровождения определяются соглашением с пользователем.

2 Термины и подготовка пользователя

Термин	Значение
Событие	Входной JSON с устойчивым id, type и data
Запуск	Сохраненная обработка события с отдельным идентификатором
Эффект и квитанция	Зафиксированное решение и результат конкретного действия
Tenant	Изолированная область данных, определяемая сервисной учетной записью
Профиль поведения	Версия модели, инструкций и знаний, допускаемая после оценки
Черновик	Сохраненный ответ, который не отправлен гостю и требует проверки

Администратор должен уметь работать с Linux, Docker, Bash, JSON, сертификатами и резервными копиями. Интегратор должен понимать HTTP, коды ошибок и повторяемость запросов. Утверждение сведений требует знания предметной области, а не только доступа к терминалу.

3 Роли права и согласования

Участник или учетная запись	Возможности и границы
Локальный администратор Docker	Установка, остановка, резервирование, запуск служебных команд. Docker-доступ эквивалентен административному доступу к серверу
Сервисная учетная запись	События и чтение разрешенных запусков, эффектов, активности; tasks.create и knowledge:read для двух агентов
hub_app	Runtime-роль PostgreSQL без SUPERUSER/BYPASSRLS; ограниченные права на рабочие таблицы
hub_publisher	Отдельная административная роль для версий поведения и одобрения фактов, не обычный HTTP-пользователь
Проверяющий сотрудник	Проверяет содержание черновика во внешнем организационном процессе. Экран согласования и отправки в комплекте отсутствует

Первичный токен не получает runs:replay. Для повторного запуска из очереди ошибок требуется отдельно выданное разрешение и причина. Нельзя предоставлять publisher-пароль гостю или обычному API-клиенту.

Требование согласования политики, публикация профиля поведения и проверка готового черновика являются разными операциями. API этой поставки не содержит универсального маршрута approve для внешнего действия. Платежи, скидки, отмены и выдача доступа не исполняются установленными адаптерами.

4 Запуск и демонстрационные сценарии

Используйте каталог и HUB_PROJECT, установленные по инструкции. После успешной первичной активации:

```
bash hub.sh status
bash hub.sh agents
bash hub.sh task "Починить замок" demo-room repair-001
```

Образец структуры ответа task, сокращенный до значимых полей:

```
{
  "eventId": "repair-001",
  "status": "task_created",
  "message": "Задача создана. Мастер не назначался.",
  "receipt": {
    "actionId": "repair-001:task",
    "actionType": "create_task",
    "status": "succeeded",
    "output": {"taskId": "UUID", "title": "Починить замок"}
  }
}
```

UUID в примере заменяется фактическим идентификатором. При повторе той же команды используется та же задача; изменение названия при прежнем repair-001 приводит к конфликту. Для нового обращения используйте новый id. Реальный исполнитель не назначается.

```
bash hub.sh ask "До какого времени мне нужно освободить квартиру?"
```

Для demo-room ожидаемый содержательный результат после успешной активации: «Выезд до 11:00.». В ответе присутствуют sourceIds, needsAdministrator, deliveryStatus: draft, reviewStatus: required и sourceValidation. Даже при needsAdministrator: false отправка не выполняется. Не выдавайте состояние draft за состоявшуюся коммуникацию.

5 HTTP API

Рекомендуемый путь первичного использования: hub.sh, который сам читает закрытые учетные данные и проверяет TLS. Для внешней интеграции локальный администратор должен безопасно предоставить клиенту CA и ограниченный токен; автоматического экспорта готового клиентского curl-config в этой версии нет.

Следующие команды являются проверяемыми примерами контракта. Файл /secure/agent-hub-http.conf заранее создается администратором вне комплекта с правами только владельцу. В нем задаются cacert и Authorization. Реальные значения токена в документ не включаются и в историю shell не вводятся.

```
curl --config /secure/agent-hub-http.conf \
  https://localhost:44318/health
curl --config /secure/agent-hub-http.conf \
  https://localhost:44318/v1/agents
curl --config /secure/agent-hub-http.conf \
  -H 'Content-Type: application/json' \
  --data @- \
  https://localhost:44318/v1/agents/maintenance-intake/runs <<'JSON'
{
  "id": "repair-api-001",
  "type": "maintenance.requested",
  "data": {"subjectId": "demo-room", "title": "Починить замок"}
}
JSON
```

Последний блок передает JSON через стандартный ввод curl. Ответ /health: {"status":"ok","service":"agent-hub-api"}. POST /runs возвращает HTTP 202 и run.id. После этого клиент

запрашивает GET /v1/runs/{id}, подставив реальный UUID из ответа. Только итоговая квитанция подтверждает результат; completed не является доказательством того, что физический ремонт выполнен.

Маршрут	Назначение и доступ
GET /health	Доступность API; без Bearer
GET /openapi.json	Схема API; без Bearer
GET /ready	Готовность базы; требуется Bearer, не проверяет LLM
GET /v1/agents	Доступные учетной записи агенты
POST /v1/agents/{agent}/runs	Асинхронный прием события
GET /v1/runs/{id}	Состояние запуска; runs:read
GET /v1/effects/{id}	Результат эффекта; effects:read
GET /v1/activity?limit=20	Активность и nextCursor; activity:read
POST /v1/runs/{id}/replay	Повтор допустимого задания с reason; отдельное runs:replay

В source/openapi.json находится экспорт контракта; актуальный работающий API предоставляет /openapi.json. Также есть синхронные маршруты /events, но для долгих операций следует использовать /runs. При ограничениях HTTP клиент не должен бесконечно повторять запрос с новым id.

6 Знания и изменение поведения

Источник знаний должен иметь привязку к объекту, версии, разрешенному агенту и сроку действия. Одновременно выбираются не более пяти одобренных записей; превышение лимита вызывает отказ, а не молчаливое отбрасывание части фактов. В установленном примере нет полного паспорта квартиры.

Типизированные факты обслуживает административный examples/agent-hub-facts.mjs: candidate, inspect, approve, revoke. Он требует закрытого publisher-подключения. Операции одобрения ссылаются на точные id, version и contentHash, содержат reviewer и reason. Измененные, просроченные и неодобренные сведения не должны использоваться как подтвержденные.

У hub.sh нет команды редактирования знаний. Для собственной интеграции администратор запускает указанный CLI в подготовленном операторском окружении; публичной формы загрузки, массового импорта PDF и автоматического обучения по переписке нет. Прямое изменение рабочих таблиц не является пользовательским сценарием.

examples/agent-hub-behavior.mjs поддерживает candidate, evaluate, publish, current и rollback. Последний публикует ранее проверенную версию поведения при выполнении того же барьера, а не откатывает схему базы. Изменение профиля или модели требует новой оценки. Поля стиля этой версии не управляют свободной генерацией ответа.

6.1 Адаптация под проект заказчика

Перед адаптацией определяются бизнес-процессы, термины, источники данных, права доступа и критерии качества. На этой основе разрабатываются конфигурация агентов, типы фактов, интеграции и предметный набор проверок. Инструкции и параметры LLM настраиваются под задачи проекта; при необходимости выполняется отдельное дообучение на согласованном наборе данных.

Обучение выполняется в среде разработки и подготовки модели. В эксплуатационный контур передаются идентифицированная версия модели и согласованные компоненты проекта. Для измененного профиля выполняются оценка и публикация по установленной процедуре. Команды базового примера не заменяют отраслевые инструкции внедренного проекта.

7 Аудит и ошибки

Хранятся события, запуски, эффекты, trace spans, квитанции, версии и результаты оценок. API активности доступен в пределах разрешенных tenant и агентов. События безопасности содержат служебные идентификаторы и статус маршрута, но не должны выводить токены и тела запросов.

Ответ или состояние	Действие оператора
400 invalid_request	Проверить схему, тип события и лишние поля; не передавать tenantId в тело
401 unauthorized	Проверить корректность, enabled и срок токена
403 forbidden	Проверить разрешения и доступный агент
409 idempotency_conflict	Сверить исходный запрос; не менять содержание прежнего события
429 rate_limited или execution_busy	Уменьшить темп; учитывать Retry-After, если он возвращен
503 operation_unconfirmed	Не считать операцию успешной; прочитать запуск и диагностику
failed или dead_letter	Разобрать причину; повторять только с полномочиями и учетом идемпотентности

Отдельно проверяйте истечение 30-дневного технического токена. Перевыпуск выполняет администратор средствами управления сервисными учетными записями. Смена часов или ослабление авторизации не являются допустимой заменой административного перевыпуска.

8 Резервное копирование и восстановление

Каталог резервных копий должен поддерживать жесткие ссылки, например ext4; FAT/exFAT не подходят. Итоговое имя копии создается только после успешной архивации, при сбое временный файл удаляется.

Для локального администратора предусмотрены команды ниже. Каталог /srv/private-backups является примером: он должен существовать, быть закрытым и находиться вне дистрибутива. Имя нового файла не должно уже существовать.

```
bash hub.sh backup /srv/private-backups/hub-20260917.tar
bash hub.sh stop
HUB_PROJECT=staysmart-agent-hub-restored \
  bash hub.sh restore /srv/private-backups/hub-20260917.tar
```

backup останавливает сервисы и сохраняет четыре тома вместе, затем пытается запустить их обратно. Копия содержит базу, пароли, токены и СА; это секретный материал. Ее нельзя публиковать или добавлять к публичному комплекту. Шифрование носителя и хранение копий обеспечивает инфраструктура организации.

restore допустим только для нового проекта без контейнеров, томов и маркера установки. Старый проект нужно остановить для освобождения порта 44318. Установщик проверяет пути и типы файлов архива. Восстанавливать можно только доверенную копию: техническая проверка архива не заменяет проверку ее происхождения.

После восстановления запускается check. Если в копии нет опубликованного профиля, токен истек или версия несовместима, проверка не пройдет. Копирование томов не обновляет сроки доступа. Периодичность резервирования и допустимые сроки восстановления определяет администратор с учетом требований организации.

9 Завершение работы и поддержка

```
bash hub.sh stop
```

Не удаляйте тома для обычной остановки. Передайте поддержке версию, обезличенный requestId/runId, статус и описание воспроизведения. Вопросы о документации направляются на info@tobesmart.ai. Режим поддержки и срок реакции определяются отдельным соглашением.