Ай-ТИ Фреш

ТЕХНИЧЕСКИЙ РАЗБОР

Локальный ИИ-контур в компании: LLM, RAG и MCP без утечки данных

Разворачиваем Ollama, Open WebUI и Qdrant на сервере клиента и подключаем ИИ к 1С через MCP

Июль 2026

itfresh.ru · ИТ-аутсорсинг для юридических лиц

О ЧЁМ ЭТОТ РАЗБОР

Суть проблемы

Сотрудники уже пользуются нейросетями — и отправляют в публичные чат-боты договоры, клиентские базы и выгрузки из 1С. Запрет не работает, а стихийное использование грозит утечкой коммерческой тайны и персональных данных. Мы решаем это управляемым ИИ-контуром внутри компании: локальная LLM, RAG по базе знаний и контролируемый доступ агента к 1С — данные не покидают периметр.

Почему это важно бизнесу

- Данные в публичных нейросетях неконтролируемы: один вставленный договор — и коммерческая тайна вне периметра компании
- Утечка персональных данных клиентов в облачный сервис — риск по 152-ФЗ плюс репутационные потери
- Рутина (выборки из 1С, поиск по регламентам, черновики писем) съедает часы специалистов — ИИ снимает её за секунды
- Локальный контур — фиксированная стоимость железа вместо растущих счетов за токены облачных API
- Запрет ИИ загоняет использование в тень; управляемый запуск даёт контроль и преимущество

ЧТО МЫ ЗАКЛАДЫВАЕМ

Ключевые параметры реализации

0.32.x

ветка Ollama на сервере клиента: единый бинарь, REST API на :11434, автоподхват GPU

Ollama, август 2026

Q4_K_M

квантизация GGUF для моделей 7–14B: укладываемся в 12–16 ГБ VRAM без заметной потери качества

наш стандарт

4

OLLAMA_NUM_PARALLEL: параллельные запросы к модели; каждый слот резервирует свой KV-кэш в VRAM

по докам Ollama

30 мин

OLLAMA_KEEP_ALIVE вместо дефолтных 5 мин: модель не выгружается между обращениями днём

наш стандарт

x4

scalar-квантизация int8 в Qdrant: сжатие RAG-индекса в 4 раза при минимальной потере точности

по докам Qdrant

2025-11-25

действующая ревизия спецификации MCP: Streamable HTTP, OAuth-авторизация и асинхронные Tasks —…

modelcontextprotocol.io

РЕАЛИЗАЦИЯ 1 ИЗ 3

Локальный LLM-сервер: Ollama на выделенном узле

Что настраиваем

Выделенная VM или GPU-узел с 16 ГБ VRAM (класс RTX 4060 Ti 16 ГБ / RTX A4000) в серверной клиента; сервис доступен только из LAN

Как мы это делаем

1. Ставим Ollama 0.32.x, в systemd-override фиксируем OLLAMA_HOST=127.0.0.1:11434, OLLAMA_KEEP_ALIVE=30m, OLLAMA_NUM_PARALLEL=4, OLLAMA_MAX_LOADED_MODELS=2
2. Подбираем модель под VRAM: 7–14B в кванте Q4_K_M; контекст num_ctx 16384 — считаем KV-кэш заранее, а не оставляем дефолт
3. Публикуем API только через nginx reverse-proxy: TLS, аутентификация, allow-list офисных подсетей; порт 11434 наружу закрыт
4. Снимаем метрики в Zabbix: время первого токена, загрузка VRAM через nvidia-smi, длина очереди; алерт при деградации

РЕЗУЛЬТАТ

Сотрудники получают ChatGPT-подобный сервис внутри офиса: документы и переписка не уходят во внешние API, скорость ответа предсказуема, стоимость фиксирована — железо вместо помесячной оплаты токенов.

КЛЮЧЕВОЙ НЮАНС

Главный расчёт — VRAM: веса модели плюс KV-кэш, растущий с num_ctx и числом слотов OLLAMA_NUM_PARALLEL. При переполнении слои тихо съезжают на CPU, и скорость падает в разы.

РЕАЛИЗАЦИЯ 2 ИЗ 3

RAG-бот по базе знаний: Open WebUI + Qdrant

Что настраиваем

Docker Compose на том же узле: Open WebUI (чат, роли, Knowledge-коллекции) + Qdrant (векторный индекс), доступ по группам

Как мы это делаем

1. Разворачиваем Open WebUI и Qdrant в docker compose; документы грузим в Knowledge-коллекции, извлечение текста — Docling/Tika с OCR для сканов
2. Эмбеддинги — многоязычная модель bge-m3: русский обязателен; модель фиксируем в конфиге — её смена требует полной переиндексации коллекций
3. Включаем гибридный поиск (ENABLE_RAG_HYBRID_SEARCH): BM25 + вектор, сверху reranker CrossEncoder, порог релевантности и ограниченный top_k
4. В Qdrant включаем int8-квантизацию (сжатие x4) и настраиваем HNSW: m и ef_construct под баланс точности и памяти
5. В системный промпт зашиваем правило: нет релевантных фрагментов — ответ «в базе знаний этого нет», без домыслов

РЕЗУЛЬТАТ

Бот отвечает по регламентам, договорам и инструкциям компании со ссылкой на документ-источник; новички находят ответы сами, а не дёргают коллег. Ни документы, ни вопросы не покидают сервер компании.

КЛЮЧЕВОЙ НЮАНС

Качество RAG решается до модели — на извлечении текста и чанкинге. Проверяем чанки глазами на выборке из 20–30 документов: один битый PDF-скан портит выдачу целой коллекции.

РЕАЛИЗАЦИЯ 3 ИЗ 3

ИИ-агент в 1С через MCP-сервер

Что настраиваем

Наш MCP-сервер поверх OData/HTTP-сервисов опубликованной базы 1С 8.3; транспорт stdio или Streamable HTTP

Как мы это делаем

1. Публикуем базу 1С на веб-сервере, включаем OData-состав только для нужных объектов; агенту заводим отдельного пользователя 1С с правами «только чтение»
2. Разворачиваем MCP-сервер по спецификации 2025-11-25: инструменты health, list_metadata, execute_query — все выборки с пагинацией top/skip
3. Режем whitelist инструментов: чтение справочников и отчётов — да; запись и проведение документов — только после обкатки и с подтверждением
4. Каждый вызов пишем в журнал: инструмент, параметры, пользователь, время; первую неделю разбираем журнал ежедневно

РЕЗУЛЬТАТ

Руководитель спрашивает бота «дебиторка по клиенту X» и получает цифры из живой базы за секунды; бухгалтерия избавлена от рутинных выборок. При этом агент физически не может изменить данные — права ограничены на уровне 1С.

КЛЮЧЕВОЙ НЮАНС

Права режем в 1С, а не промптом: системный промпт — не защита. Лимиты выборки обязательны, иначе агент утащит в контекст справочник на сто тысяч позиций и «съест» окно модели.

ЧТО МЫ ВЫУЧИЛИ НА ПРАКТИКЕ

Подводные камни

× **Модель выбрана без расчёта VRAM**
Веса + KV-кэш × num_ctx × слоты параллелизма. При переполнении Ollama тихо выгружает слои на CPU — скорость падает в разы. Считаем бюджет заранее

× **Порт 11434 открыт наружу**
У Ollama нет встроенной аутентификации. Держим API на 127.0.0.1, наружу — только nginx с TLS, авторизацией и allow-list подсетей

× **Дефолтный OLLAMA_KEEP_ALIVE=5m**
Модель выгружается между обращениями, и каждый первый запрос ждёт холодную загрузку. В рабочие часы ставим 30m и больше под профиль нагрузки

× **RAG по сырым PDF-сканам**
Извлечение без OCR даёт пустые чанки — бот «не видит» документов. Прогоняем через Docling/Tika с OCR и проверяем чанки на выборке

× **Эмбеддинги без русского языка**
Англоязычная embedding-модель промахивается на русских запросах. Берём многоязычную bge-m3 и фиксируем: смена модели = полная переиндексация

× **Нет порога релевантности**
Без score threshold бот сочиняет ответ из нерелевантных чанков. Ставим порог, ограничиваем top_k, в промпт — правило честного «не знаю»

× **ИИ-агент под админской учёткой 1С**
Системный промпт — не защита: агент сможет менять данные. Отдельный пользователь «только чтение», whitelist инструментов MCP, журнал вызовов

× **Запуск сразу на всю компанию**
Без пилота — вал жалоб на качество ответов. Запускаем на одном отделе, две недели собираем вопросы-провалы и дообогащаем базу знаний

УРОВНИ РЕАЛИЗАЦИИ

Как правильно

МИНИМУМ

- Политика ИИ: что можно отправлять в облачные модели, что — только в локальный контур
- Ollama на 127.0.0.1, доступ только через reverse-proxy с TLS и аутентификацией
- Пилот: модель 7–8B в Q4_K_M на существующем железе, один отдел на две недели

НОРМАЛЬНО

- Выделенный GPU-узел от 16 ГБ VRAM; KEEP_ALIVE и NUM_PARALLEL под профиль нагрузки
- Open WebUI: группы доступа, RAG с гибридным поиском BM25+вектор и реранкером
- Мониторинг в Zabbix: время первого токена, VRAM, очередь запросов, алерты
- Бэкап коллекций Qdrant и конфигов наравне с остальной инфраструктурой

ХОРОШО

- MCP-агенты в 1С/CRM: read-only учётки, whitelist инструментов, журнал вызовов
- Двухконтурная схема: конфиденциальное — локально, нечувствительное — облачные API
- Порог релевантности и reranker CrossEncoder против галлюцинаций в RAG
- Квартальный прогон эталонных задач на новых моделях — апгрейд по факту, не по хайпу

ПРОВЕРЬТЕ У СЕБЯ

Чек-лист самопроверки

- [ ] Есть ли письменная политика: какие данные сотрудникам можно отправлять в облачные нейросети?
- [ ] LLM-API (Ollama, порт 11434) закрыт от интернета и доступен только через прокси с аутентификацией?
- [ ] Посчитан ли VRAM-бюджет: веса модели плюс KV-кэш на выбранный num_ctx и число параллельных запросов?
- [ ] Эмбеддинг-модель поддерживает русский язык и зафиксирована в конфиге RAG?
- [ ] Стоит ли порог релевантности, чтобы бот отвечал «в базе этого нет» вместо выдумок?
- [ ] У ИИ-агента в 1С отдельная учётка с минимальными правами, а не логин администратора?
- [ ] Ведётся ли журнал вызовов инструментов агента: кто, что и когда запросил?
- [ ] Бэкапится ли векторная база и конфигурация ИИ-контура наравне с 1С и файлами?
- [ ] Есть ли план отката: как за минуты отключить ИИ-агента, не сломав рабочие процессы?

Если хотя бы на два вопроса ответ «нет» или «не знаю» — тема требует внимания.

ЧТО МЫ ДЕЛАЕМ

Как поможет ITFresh

ITFresh — ИТ-аутсорсинг для юридических лиц до 50 рабочих мест в Москве и области. 15+ лет практики, собственная инфраструктура в дата-центре МТС (8 серверов Dell Xeon Platinum).

- Разворачиваем локальный ИИ-контур под ключ: подбор GPU, Ollama, Open WebUI, Qdrant, TLS-доступ и мониторинг
- Строим RAG-бота по вашей базе знаний и регламентам — документы не покидают сервер компании
- Подключаем ИИ-агентов к 1С через собственный MCP-сервер: права, whitelist, аудит, боевая обкатка
- Пишем политику использования нейросетей и обучаем сотрудников безопасной работе с ИИ
- Сопровождаем контур: обновление моделей, контроль качества ответов, метрики VRAM и latency в Zabbix

15+
лет в ИТ-поддержке

50
рабочих мест — наш профиль

МТС
дата-центр, Москва

КОНТАКТЫ

Обсудить вашу задачу

Сайт	**itfresh.ru**
Телефон	**+7 903 729-62-41**
Telegram	**@ITfresh_Boss**

Бесплатно посмотрим вашу инфраструктуру по этому чек-листу и скажем, где тонко — без обязательств.

itfresh.ru

ДОКУМЕНТАЦИЯ И НАШ СТЕК

Техническая база

01 Ollama — FAQ и переменные окружения (docs.ollama.com — 2026)

02 Open WebUI — RAG, hybrid search, Knowledge (docs.openwebui.com — 2026)

03 Qdrant — Quantization, HNSW, Optimize Performance (qdrant.tech — 1.x)

04 MCP Specification — транспорты, авторизация, tools (modelcontextprotocol.io — 2025-11)

05 Шаблон ITfresh: systemd-юнит Ollama + nginx-прокси с TLS (itfresh.ru — 2026)

06 Шаблон ITfresh: политика использования нейросетей (itfresh.ru — 2026)

Основано на официальной документации продуктов и нашей практике внедрения.