

Развёртывание OpenClaw для бизнеса

От локального агента к безопасной, надёжной и управляемой бизнес-инфраструктуре.

РЕЗЮМЕ

Краткое резюме

OpenClaw привлекателен тем, что необычайно открыт. Он даёт среду выполнения агентов, размещаемую на собственной инфраструктуре, не привязывая организацию к одному облаку, одному поставщику моделей, одной системе идентификации или одной операционной модели. Эта гибкость и есть ключевое свойство — и одновременно источник нагрузки при развёртывании.

В корпоративном применении работа быстро выходит за рамки установки. Организации предстоит создать устойчивую площадку размещения, определить границы идентификации и учётных данных, подключить модели и бизнес-системы, ограничить возможности агентов, защититься от специфичных для агентов угроз — таких как внедрение промптов и чрезмерная автономность — и выстроить мониторинг, обновления и восстановление.

Вывод практический: организациям, выбирающим OpenClaw, следует рассматривать окружающую среду как часть системы. Продуманная модель развёртывания делает решения по инфраструктуре, безопасности и эксплуатации явными, автоматизирует то, что можно повторить, документирует то, что остаётся специфичным для конкретной нагрузки, и обеспечивает возможность пересоздать среду.

OpenClaw даёт среду выполнения. Готовность к промышленной эксплуатации обеспечивают архитектура, средства контроля и эксплуатационные практики вокруг неё.

Четыре вывода

- 01 OpenClaw убедителен**, когда владение средой выполнения, гибкость выбора моделей и расширяемость важнее полностью управляемой SaaS-модели эксплуатации.
- 02 Управляемые агентные платформы — сильная альтернатива**, когда организация хочет передать поставщику большую часть нагрузки по платформе, среде выполнения и эксплуатации.
- 03 Готовность к эксплуатации зависит** от идентификации, разрешений, защиты от внедрения промптов, наблюдаемости и восстановления не меньше, чем от самой среды выполнения агентов.
- 04 Настоящая веха — не** «у нас запущен OpenClaw», а «мы можем предсказуемо развёртывать, контролировать, эксплуатировать и пересоздавать его».

01 Почему OpenClaw важен для бизнеса

Важный сдвиг в корпоративном ИИ не в том, что модели лучше отвечают на вопросы. Он в том, что агенты могут быть постоянно доступны, использовать инструменты, сохранять контекст и выполнять действия в разных системах. OpenClaw объединяет в одной среде выполнения самостоятельно размещаемый gateway, каналы, сессии, память, инструменты, навыки и маршрутизацию между агентами.^[1]

ВЫГОДА ДЛЯ БИЗНЕСА

ПОЧЕМУ ЭТО ВАЖНО

Постоянно доступные агенты

Постоянно работающая среда выполнения может обслуживать процесс или команду, а не зависеть от того, откроет ли один сотрудник окно чата.

Действия в бизнес-системах

Инструменты и подключения MCP позволяют агенту перейти от ответов на вопросы к чтению данных и действиям в CRM, ERP, Microsoft 365 и API.

Гибкость выбора моделей

Среду выполнения можно настроить на нескольких поставщиков моделей и сценарии переключения при отказе, вместо того чтобы делать одну модель самим приложением.^[1]

Контроль над средой выполнения

Организация решает, где размещается среда выполнения, как она расширяется и какой частью окружающей архитектуры организация владеет.

Агенты, выстроенные вокруг работы

У большинства компаний информации уже достаточно; трудность — собрать её из разных систем и затем сделать следующий шаг. Агент может стать связующим слоем, который собирает контекст, рассуждает над ним и использует инструменты, чтобы продвинуть работу. Это не то же самое, что просто выдать каждому сотруднику персонального чат-бота.

Для долгоживущих бизнес-агентов модель всё чаще можно рассматривать как инфраструктуру под агентом, а не как само приложение.

02 Место OpenClaw

Альтернативы OpenClaw — не прямые аналоги: они находятся на разных уровнях. Полезнее спрашивать, какая операционная модель лучше отвечает характеру работы, требованиям к контролю, потребностям в интеграции и инженерным возможностям организации.

ВАРИАНТ	КАТЕГОРИЯ	СИЛЬНЫЕ СТОРОНЫ	КОГДА ПОДХОДИТ ЛУЧШЕ ВСЕГО
Copilot Studio	Управляемая low-code платформа для агентов	Разработка, коннекторы, управление и публикация в экосистеме Microsoft	Когда организации нужна управляемая платформа и минимум ответственности за среду выполнения. ^[2]
Microsoft Foundry Agent Service	Управляемый хостинг агентов	Промпт-агенты и размещённые агенты с управляемыми конечными точками, масштабированием, идентификацией и наблюдаемостью	Когда нужен собственный код, но предпочтительны хостинг и корпоративные средства контроля от Microsoft. ^[3]
OpenClaw	Самостоятельно размещаемая среда выполнения агентов	Владение средой выполнения, каналы, инструменты, навыки, гибкость моделей, открытая архитектура	Когда организация хочет владеть средой выполнения и решениями по окружающей архитектуре. ^[1]
LangGraph / CrewAI	Фреймворки для разработчиков	Максимальный контроль над кодом агентных приложений и оркестрацией	Когда команда создаёт заказное агентное приложение, а не разворачивает готовую среду выполнения.
Zapier / аналоги	Управляемая ИИ-автоматизация	Быстрая связность с SaaS и автоматизация рабочих процессов	Когда широта SaaS-автоматизации важнее владения средой выполнения.

Управляемые платформы: когда меньше владения средой выполнения — лучше

Иногда управляемая корпоративная платформа — более удачный выбор. Copilot Studio силён там, где на первом месте low-code разработка, эксплуатация силами Microsoft и встроенные средства управления. OpenClaw убедительнее, когда организация ценит владение средой выполнения, широкую расширяемость, гибкость выбора моделей и возможность формировать окружающую архитектуру. Вопрос не в том, какой продукт лучше, а в том, какой операционной моделью организация хочет владеть.

Управляемый хостинг и фреймворки для разработчиков

Управляемые сервисы хостинга агентов полезны, когда команда разработки хочет писать собственный код, а поставщик берёт на себя большую часть хостинга, идентификации и наблюдаемости. Фреймворки для разработчиков, такие как LangGraph и CrewAI, находятся ниже в стеке и уместны, когда команда намерена строить само агентное приложение.

03 Когда OpenClaw — неверный выбор

Убедительное архитектурное решение включает и критерии отказа. OpenClaw не становится лучшим ответом только потому, что он открыт или что среда выполнения мощная.

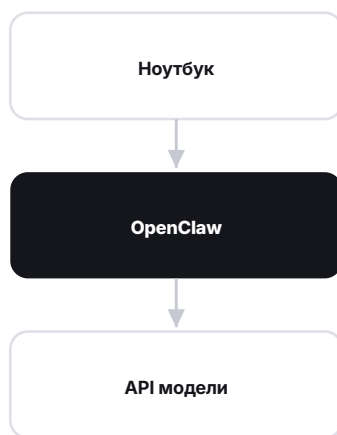
КРИТЕРИЙ ОТКАЗА	БОЛЕЕ УДАЧНОЕ РЕШЕНИЕ
Вам нужны управляемая поставщиком среда выполнения и SLA	Управляемая платформа, например Copilot Studio или Foundry Agent Service, может лучше подойти по модели эксплуатации.
Задача — детерминированная автоматизация рабочих процессов	Power Automate, Logic Apps, Zapier или другой инструмент автоматизации может оказаться проще и удобнее в управлении.
Никто не отвечает за облачную эксплуатацию	Самостоятельно размещаемая среда выполнения порождает обязанности по обновлениям, мониторингу, восстановлению и безопасности. Если за них никто не отвечает, не создавайте такое обязательство.
Жёсткие регуляторные требования, которые ваша команда не может выполнить	Выберите платформу, чьи управляемые средства контроля, сертификации, модель поддержки и возможности аудита соответствуют требованию.
Организации не нужны ни владение средой выполнения, ни гибкость выбора моделей	Главные преимущества OpenClaw могут не оправдать эксплуатационную нагрузку.
Сценарий использования — один узкий ассистент	Управляемый агент или встроенная функция приложения могут оказаться существенно проще универсальной среды выполнения.

Открытый код — не стратегия. Самостоятельное размещение оправдано лишь тогда, когда получаемый контроль стоит возникающей ответственности.

04 От локального агента к бизнес-инфраструктуре

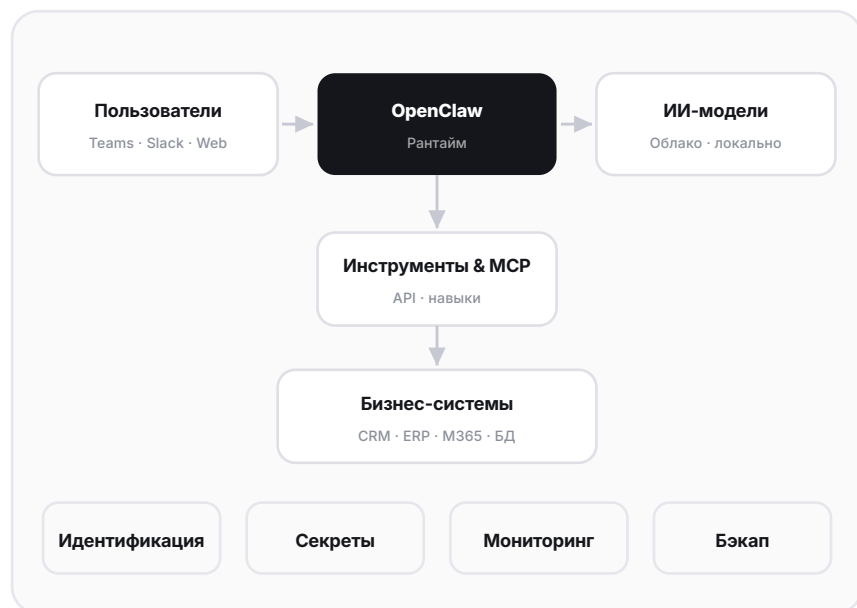
OpenClaw намеренно делает первый опыт простым. Это отлично для разработки и экспериментов. Корпоративное развёртывание начинается на уровень раньше: сначала организации нужно устойчивое окружение, в котором сможет работать среда выполнения.

ЛОКАЛЬНЫЙ ТЕСТ



Быстро попробовать · Легко менять
Владелец рядом

БИЗНЕС-РАЗВЁРТЫВАНИЕ



Постоянные вычисления · сеть · эксплуатация · управление

Облачная VM — часто самый простой вариант, но постоянные вычислительные мощности сразу порождают выбор: регион, операционная система, хранилище, администрирование, поведение при перезапуске, обновления, мониторинг и восстановление.

Вопрос разработчика — «смогу ли я это запустить?».
Вопрос бизнеса — «сможем ли мы создать окружение, где это работает надёжно, непрерывно и под известными средствами контроля?».

05 Базовый уровень промышленного развёртывания

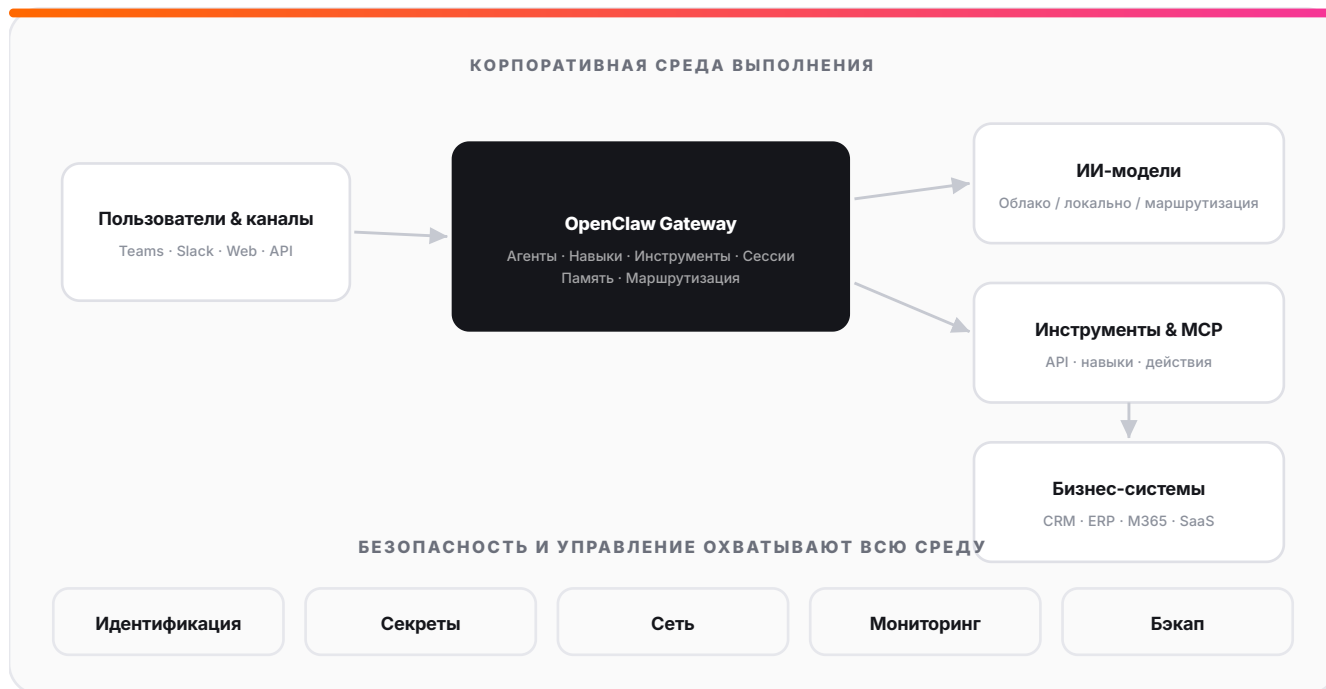
Полезное руководство по развёртыванию не должно ограничиваться вопросами. Этот базовый уровень фиксирует средства контроля и эксплуатационные решения, которые обычно следует принять осознанно, прежде чем считать среду OpenClaw бизнес-инфраструктурой. Конкретные продукты и поставщики могут различаться — требования нет.

РЕШЕНИЕ	РЕКОМЕНДУЕМЫЙ БАЗОВЫЙ УРОВЕНЬ	ЗАЧЕМ
Хост среды выполнения	Постоянные вычислительные мощности во владении организации	Делает среду выполнения независимой от устройства сотрудника и задаёт организации определённую границу эксплуатации.
Организационная идентификация	Выделенная идентификация рабочей нагрузки (workload identity), где она поддерживается ^[13]	Избавляет от привязки промышленного доступа к личным учётным данным того, кто установил агента.
Доступ к ресурсам	Доступ на основе удостоверений там, где платформа это поддерживает ^[13]	Снижает зависимость от долгоживущих секретов, упрощает отзыв прав и аудит.
Секреты	Управляемое хранилище секретов; никаких локальных файлов с промышленными секретами у разработчиков ^[15]	Рассматривает токены и ключи как управляемые активы с контролем доступа, ротацией и процедурами восстановления.
Доступ к моделям	Утверждённые поставщики и модели с документированной политикой по регионам, квотам и резервным вариантам ^{[10][12]}	Превращает выбор модели в эксплуатационную политику, а не в предпочтение разработчика.
Точки взаимодействия с людьми	Утверждённые корпоративные каналы, соответствующие сценарию	Оставляет доступ к агенту внутри коммуникационных сред, которыми организация может управлять.
Эксплуатация	Централизованные контроль состояния, логирование и оповещения плюс документированные процедуры перезапуска и обновления	Делает сбои видимыми и даёт операторам воспроизводимый сценарий реагирования.
Развёртывание	Скриптованная воспроизводимая конфигурация вместо собранных вручную уникальных VM ^[14]	Делает пересоздание, развёртывание второго экземпляра и восстановление реалистичными.

Это базовые архитектурные принципы, а не универсальная политика. Топология сети, конкретные разрешения, сроки хранения, пороги согласования, целевые показатели восстановления и утверждённые поставщики услуг по-прежнему зависят от нагрузки и организации.

06 Референсная архитектура OpenClaw для бизнеса

Схема — это референсный шаблон, а не архитектура, предписанная OpenClaw.



Граница среды выполнения

Окружение вокруг неё — идентификация, секреты, сеть, мониторинг и восстановление — это то, что превращает гибкую среду выполнения в эксплуатируемую систему.

Идентификация определяет, кем или чем является агент. Секреты определяют, какие учётные данные он может использовать. Сеть определяет, до чего он может дотянуться. Мониторинг определяет, увидят ли операторы сбой. Резервное копирование и восстановление определяют, можно ли вернуть среду в рабочее состояние.

OpenClaw даёт организации свободу формировать эту границу; шаблон развёртывания под Microsoft принимает определённую часть этих решений заранее.

07 Идентификация, секреты и сеть

ИДЕНТИФИКАЦИЯ

Сделайте агента субъектом организации

Не привязывайте промышленного агента к личным учётным данным сотрудника, который его установил. Используйте выделенную организационную идентификацию рабочей нагрузки, если целевая платформа её поддерживает. Для некоторых внешних систем могут по-прежнему требоваться учётные данные приложений, но они должны быть управляемыми исключениями с явным владельцем жизненного цикла, принципом наименьших привилегий и процедурами отзыва.

СЕКРЕТЫ

Сначала избавьтесь от лишних секретов

Лучший секрет — тот, который развёртыванию не нужен. Доступ на основе удостоверений сокращает число «сырых» учётных данных, которые приходится хранить. Там, где секреты всё же необходимы, размещайте их в управляемом хранилище секретов, ограничивайте доступ идентификацией среды выполнения и вводите документированные процедуры ротации и реагирования на инциденты. Избегайте зашитых в код учётных данных и локальных файлов секретов у разработчиков в промышленной среде.

СЕТЬ

Сетевая достижимость — это политика

Промышленный дизайн должен осознанно задавать входящий административный доступ и исходящую достижимость, а не наследовать удобные настройки среды разработки. Предпочтительно не иметь лишнего публичного входящего пути; используйте контролируемый административный маршрут — частное подключение, bastion-сервис, VPN или эквивалент. Исходящий доступ следует ограничить конечными точками моделей, каналами, источниками пакетов и бизнес-системами, которые нагрузке действительно нужны.

Идентификация определяет, кто такой агент. Секреты определяют, какие двери он может открыть. Сеть определяет, до каких дверей он может дойти.

08 Интеграция с бизнес-системами — там, где встречаются ценность и риск

Ценность OpenClaw резко возрастает, когда агент может работать с CRM, ERP, средами совместной работы, базами данных, API и MCP-серверами. Возрастает и последствия неверного решения. Наличие инструмента не заменяет политику.

ВОЗМОЖНОСТЬ	ПРИМЕР	РЕКОМЕНДУЕМЫЙ КОНТРОЛЬ
Чтение	Получение контекста по клиенту, заказу или документу	Принцип наименьших привилегий; область данных определяется потребностью пользователя или процесса.
Анализ	Обобщение, классификация, сравнение, выявление отклонений	Сохраняйте исходный контекст и не считайте сгенерированную интерпретацию достоверными данными без указания их происхождения.
Рекомендация	Подготовка ответа или предложение следующего шага	Определите, когда перед выполнением требуется проверка человеком.
Запись	Обновление CRM или создание задачи	Ограничьте изменяемые объекты и поля и используйте идентификацию, позволяющую атрибутировать действие.
Выполнение	Запуск API, рабочего процесса или бизнес-процесса	Определите предусловия, лимиты, идемпотентность и точки согласования.
Внешнее действие	Отправка письма, внешняя публикация, изменение состояния, видимого клиенту	Требуйте более строгого согласования, доказательств для аудита и возможности быстрого отзыва прав.

Цель — не максимальная автономность везде. Цель — правильная автономность для конкретной работы.

09 Специфичные для агентов угрозы: внедрение промптов и запутанный заместитель

Одни только разрешения не делают агента безопасным. Агент может быть законно уполномочен использовать инструмент — и всё равно быть склонён к неправильному применению этих полномочий. NIST рассматривает прямое и косвенное внедрение промптов как риск для генеративных ИИ-систем, включая атаки, встроенные в контент, который приложение с LLM позднее извлекает.^[7] OWASP относит внедрение промптов (Prompt Injection) и чрезмерную автономность (Excessive Agency) к главным рискам LLM-приложений.^[8]

УГРОЗА	КАК ЭТО ВЫГЛЯДИТ	СХЕМА КОНТРОЛЯ
Прямое внедрение промптов	Пользователь указывает агенту игнорировать политику или использовать инструмент не по назначению	Принудительное применение политик на системном уровне, разрешённый список инструментов, ограниченные по области удостоверения, согласование значимых действий.
Косвенное внедрение промптов	Вредоносные инструкции, скрытые в веб-странице, документе, письме или полученных данных	Считайте полученный контент недоверенными данными; отделяйте контент от управляющих инструкций; ограничивайте вызовы инструментов и исходящий трафик.
Поведение «запутанного заместителя»	У агента есть законные полномочия, но недоверенный источник вынуждает его применить эти полномочия не по назначению	Связывайте действия с аутентифицированным инициатором или процессом, проверяйте намерение и контекст, требуйте согласования там, где последствия существенны.
Чрезмерная автономность	У агента больше инструментов, прав или самостоятельности, чем требует задача	Принцип наименьших привилегий, узкие схемы инструментов, явные границы записи и выполнения, по возможности короткоживущие учётные данные.
Небезопасная обработка вывода	Сгенерированный вывод напрямую передаётся в код, SQL, shell, HTML или другую систему	Проверяйте и очищайте вывод; по возможности используйте типизированные интерфейсы инструментов вместо произвольного выполнения команд.

Безопасная архитектура агента должна управлять и тем, что агенту разрешено делать, и тем, какая информация может влиять на это решение.¹¹

10 Модели, надёжность и эксплуатация

Стратегия по моделям

Локальный пользователь может выбрать любимую модель. Бизнесу нужна политика: утверждённые поставщики, география, условия конфиденциальности, задержки, квоты, поведение при переключении и контроль затрат. Microsoft Foundry может дать управляемый слой доступа к моделям, тогда как OpenClaw остаётся средой выполнения. [3][10][11] Разные задачи могут оправдывать разные модели; бизнес-процесс не должен быть жёстко привязан к одному поколению моделей.

Надёжность и наблюдаемость

Когда процесс зависит от агента, операторам нужно знать, исправен ли gateway, подключены ли каналы, не отказывает ли доступ к моделям, не истекли ли учётные данные и какие значимые действия агент пытался выполнить. OpenClaw предоставляет диагностику gateway и проверки каналов, но окружающая операционная модель всё равно требует владельца, оповещений и процедур реагирования. [6]

Сайзинг и стоимость: фиксируйте факторы затрат, а не мнимую точность

ФАКТОР ЗАТРАТ	ПРАКТИЧЕСКАЯ РЕКОМЕНДАЦИЯ
VM / вычисления	Требования к CPU и памяти зависят от каналов, параллельности, локальных инструментов и от того, работают ли модели удалённо или локально. Начните с умеренной VM общего назначения и проведите нагрузочное тестирование на реальной нагрузке, прежде чем масштабировать.
Использование моделей	Обычно основная переменная статья затрат, когда модели удалённые. Отслеживайте токены и запросы по агентам и задачам, устанавливайте квоты и направляйте простую работу на менее дорогие модели там, где это уместно.
Хранилище / логи	Состояние диалогов, логи и артефакты накапливаются. Определите сроки хранения и следите за ростом хранилища, а не считайте VM бесконечной.
Эксплуатация	Скрытая статья затрат — человеческое владение: установка обновлений, реагирование на инциденты, тестирование обновлений и пересмотр прав доступа. Автоматизация снижает эти затраты сильнее, чем экономия небольшой суммы на VM.

Точные цены здесь намеренно не приводятся: облачный регион, тип вычислений и тарифы на модели меняются. Модель развёртывания должна делать эти переменные наблюдаемыми и заменяемыми, а не скрывать их.

11 Матрица готовности к промышленной эксплуатации

Это практический внутренний контрольный рубеж. Бизнесу не нужны одинаковые средства контроля для каждой нагрузки, но решения должны быть явными.

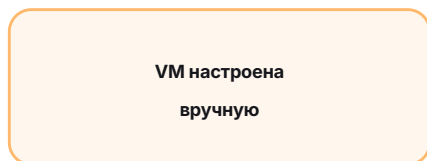
ВОЗМОЖНОСТЬ	ЭКСПЕРИМЕНТ	БИЗНЕС-РАЗВЁРТЫВАНИЕ	СОСТОЯНИЕ ГОТОВНОСТИ К ЭКСПЛУАТАЦИИ
Вычисления	Ноутбук / по ситуации	Постоянный хост	Известные сайзинг, владелец, перезапуск и восстановление
Идентификация	Личные учётные данные	Выделенная схема идентификации	Наименьшие привилегии, жизненный цикл, аудируемость
Секреты	Переменные окружения / конфиг	Защищённое хранилище	В хранилище, с ротацией и контролем доступа
Сеть	Удобные значения по умолчанию	Ограниченный входящий / исходящий трафик	Документированный админ-доступ и сегментация
Модели	Предпочитаемая модель	Утверждённый поставщик / модель	Политика по региону, квотам, резервированию и затратам
Интеграции	Токены разработчика	Ограниченный корпоративный доступ	Явные границы чтения / записи / выполнения
Защита от внедрения промптов	Обычно отсутствует	Guardrails для промптов и инструментов	Работа с недоверенным контентом, согласования и контроль исходящего трафика
Мониторинг	Ручная проверка	Проверки состояния / логи	Оповещения, владелец, процедуры реагирования
Обновления	По ситуации	Плановые	Тестирование, откат и дисциплина релизов
Восстановление	Обычно отсутствует	Бэкап VM / конфигурации	Проверенная процедура пересоздания и восстановления
Документация	Личные заметки	Запись о развёртывании	Воспроизводимое описание среды

Запущенный агент — технический результат.
Эксплуатируемый агент — бизнес-возможность.

12 Жизненный цикл и воспроизводимость — это один и тот же аргумент

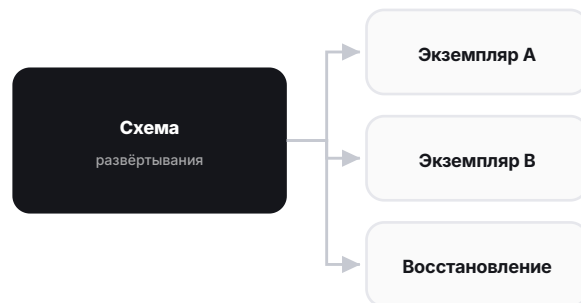
Промышленная среда не заканчивается установкой. Её нужно подготовить, защитить, настроить, подключить, проверить, эксплуатировать, обновлять и в итоге пересоздавать.

РАБОТАЮЩАЯ МАШИНА



Работает сегодня
Знание — у того, кто её собрал

ВОСПРОИЗВОДИМОЕ РАЗВЕРТЫВАНИЕ



Тот же замысел · известная конфигурация · воспроизводимый результат

Эти шаги жизненного цикла становятся надёжными только тогда, когда решения по развёртыванию зафиксированы, а не живут в памяти одного инженера.

Тест на пересоздание

Если бы эта VM исчезла сегодня ночью, смог бы другой администратор воссоздать среду завтра? Смогла бы компания развернуть второй экземпляр для другой команды или региона? Смогла бы она объяснить, какая конфигурация выбрана осознанно, а какая настройка — исторический остаток? Инфраструктура как код, скриптованная конфигурация, записи о развёртывании и контролируемые пути обновления превращают работающую машину в воспроизводимую возможность.

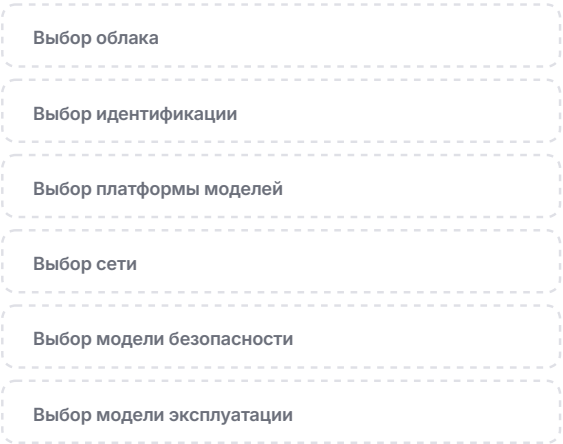
Работающая VM — это экземпляр. Воспроизводимая среда — это возможность.

13 От OpenClaw к RapidClaw

Всё сказанное выше применимо независимо от облака и поставщика. OpenClaw намеренно оставляет окружающую архитектуру открытой. RapidClaw начинается там, где эта открытость превращается в нагрузку при развёртывании для организаций, стандартизированных на Microsoft.

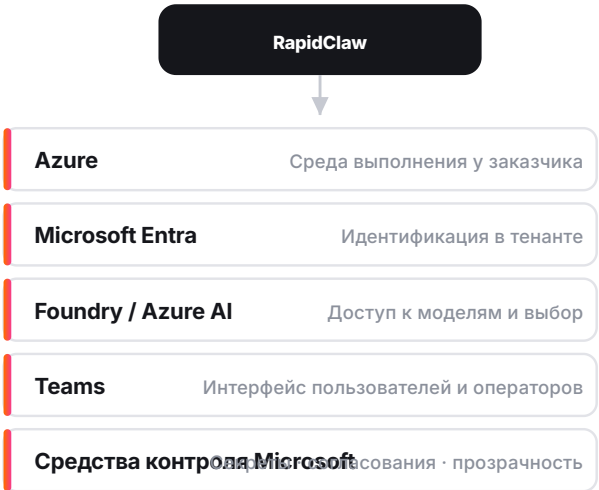
RapidClaw — не форк OpenClaw. OpenClaw остаётся средой выполнения. RapidClaw — это выработанная модель развёртывания и эксплуатации вокруг него для организаций на Microsoft. На практике это означает, что управляемое развёртывание поднимает всю среду внутри собственного тенанта Azure заказчика примерно за двадцать минут, а не в виде собранной вручную VM, конфигурация которой остаётся в голове того, кто её собрал.^[9]

OPENCLAW: ПОЛНОСТЬЮ ОТКРЫТ



Гибкость — это преимущество.
Все решения принимает организация.

RAPIDCLAW: ГОТОВЫЕ РЕШЕНИЯ



Меньше архитектурных решений · больше воспроизводимости

ОБЛАСТЬ РЕШЕНИЯ	ПОЗИЦИЯ RAPIDCLAW
Границы облака	Среда выполнения в Azure, принадлежащая заказчику ^{[9][14]}
Уровень идентификации	Организационная идентификация на базе Microsoft Entra ^{[9][13]}
Платформа моделей	По умолчанию путь Microsoft Foundry / Azure AI ^{[3][10][12]}
Точки взаимодействия с людьми	Microsoft Teams там, где это уместно
Средства контроля	Согласованные с Microsoft схемы работы с секретами, доступом, прозрачностью и управлением ^{[9][11][15]}
Развёртывание	Управляемая воспроизводимая конфигурация вместо разовой ручной сборки ^{[9][14]}

14 Что меняет RapidClaw — и что оставляет открытым

RapidClaw намеренно узок в одном измерении и открыт в другом. Он стандартизирует решения по развёртыванию в Microsoft, которые не нужно изобретать заново, оставляя сам OpenClaw открытым для агентов, навыков, инструментов и подключений к бизнес-системам, подходящих конкретной нагрузке.

ГРАНИЦА	СОДЕРЖАНИЕ
RapidClaw стандартизирует	Путь развёртывания в Azure; интеграцию с идентификацией Microsoft; путь моделей Foundry / Azure AI; интеграцию с Teams; воспроизводимую настройку и операционную модель.
Заказчик по-прежнему решает	Каких агентов создавать; какие бизнес-системы и MCP-серверы открывать; права на чтение, запись и выполнение; политику согласований; сроки хранения; средства контроля для регулируемых нагрузок; бизнес-владение.
RapidClaw не делает	Не форкает и не заменяет OpenClaw; не делает равноправными все возможные архитектуры облака и среды выполнения; не отменяет необходимость корпоративного управления.

Ценность выработанной модели развёртывания не в том, что она снимает все решения. Она снимает те решения, которые организациям на Microsoft не должно приходиться принимать с нуля каждый раз.

Заключение

Полностью открытая архитектура OpenClaw — важная часть его привлекательности. Сложное начинается тогда, когда эта гибкость должна стать надёжной бизнес-инфраструктурой: постоянные вычисления, идентификация, секреты, сеть, политика по моделям, границы интеграций, защита от внедрения промптов, наблюдаемость, обновления и восстановление.

Для организаций, которые уже работают на Microsoft, RapidClaw — наш ответ на этот разрыв в развёртывании: оставить OpenClaw как среду выполнения, один раз принять архитектурные решения по Microsoft и превратить их в воспроизводимую операционную модель.

ИСТОЧНИКИ

[1] Документация OpenClaw, docs.openclaw.ai, дата обращения: август 2026 г.

docs.openclaw.ai

[2] Microsoft Learn, «What is Microsoft Copilot Studio?», обновлено 3 августа 2026 г.

learn.microsoft.com/en-us/microsoft-copilot-studio/fundamentals-what-is-copilot-studio

[3] Microsoft Learn, «What is Microsoft Foundry Agent Service?», обновлено 19 августа 2026 г.

learn.microsoft.com/en-us/azure/foundry/agents/overview

[4] Документация LangGraph, LangChain, дата обращения: август 2026 г.

docs.langchain.com/oss/python/langgraph/overview

[5] Документация CrewAI, дата обращения: август 2026 г.

docs.crewai.com/en/introduction

[6] Документация OpenClaw, руководство по эксплуатации gateway и диагностика, дата обращения: август 2026 г.

docs.openclaw.ai/gateway

[7] NIST AI 600-1, «Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile», июль 2024 г., §2.9 Information Security.

nvlpubs.nist.gov/nistpubs/ai/NIST.AI.600-1.pdf

[8] «OWASP Top 10 for LLM Applications 2026» — внедрение промптов (Prompt Injection) и чрезмерная автономность (Excessive Agency), OWASP GenAI Security Project, август 2026 г.

genai.owasp.org/resource/owasp-genai-llm-top-10-2026

[9] RapidStart, «RapidClaw for OpenClaw — Governed AI in Azure», дата обращения: август 2026 г.

www.rapidstart.com/en/products/rapidclaw-for-openclaw

[10] Microsoft Learn, «Built-in policies for model deployment in Microsoft Foundry portal», обновлено 18 августа 2026 г.

learn.microsoft.com/en-us/azure/foundry/how-to/model-deployment-policy

[11] Microsoft Learn, «Role-based access control for Microsoft Foundry», обновлено 3 июля 2026 г.

learn.microsoft.com/en-us/azure/foundry/concepts/rbac-foundry

[12] Microsoft Learn, «Deployment types for Microsoft Foundry Models», обновлено 6 августа 2026 г.

learn.microsoft.com/en-us/azure/foundry/foundry-models/concepts/deployment-types

[13] Microsoft Learn, «Managed identities for Azure resources», документация Microsoft Entra.

learn.microsoft.com/en-us/entra/identity/managed-identities-azure-resources/overview

[14] Microsoft Learn, «What is an Azure landing zone?», Cloud Adoption Framework, обновлено 26 августа 2026 г.

learn.microsoft.com/en-us/azure/cloud-adoption-framework/ready/landing-zone

[15] Microsoft Learn, «Understanding autorotation in Azure Key Vault», обновлено 10 апреля 2026 г.

learn.microsoft.com/en-us/azure/key-vault/general/autorotation

OpenClaw — это среда выполнения. RapidClaw — это модель развёртывания Microsoft вокруг неё.

О компании RapidStart

RapidStart создаёт бизнес-приложения и агентную инфраструктуру на базе Microsoft, чтобы передовые бизнес-технологии было проще развёртывать, эксплуатировать и внедрять.