Один человек, три проекта, пять агентов: как устроен solo AI-native контур

2026-06-10

Один человек, три проекта, пять агентов: как устроен solo AI-native контур

Что на самом деле ограничивает одного оператора?

Один человек ведёт три живых проекта одновременно: клиентский внедренческий проект, продуктовый прототип на ранней стадии и собственный research-контур, который каждую ночь производит десятки отчётов. Рядом с ним работают пять агентных ролей — research agent, ops monitor, code agent и две вспомогательные. Это не демонстрация и не эксперимент на выходные, а воспроизводимый операционный паттерн, который мы наблюдаем у нескольких практиков с конца 2025 года. Solo AI-native контур — это операционная модель, в которой один человек оркестрирует несколько специализированных агентных ролей, работающих по расписанию и производящих готовые артефакты. И первое, что становится видно вблизи: ограничивает такого оператора не мощность моделей и не объём генерации. Ограничивает количество циклов внимания — сколько раз за день один человек успевает закрыть петлю «задача → артефакт → ревью». «Задача → артефакт → ревью» — это минимальный цикл агентной работы: агент получает задание, производит проверяемый результат-файл, а оператор оценивает и решает, идёт ли он в дело. Симон Уиллисон, который публично описал свой режим работы с несколькими агентами параллельно, формулирует это прямо: к одиннадцати утра он умственно вымотан, хотя машинного времени получает больше (Simon Willison).

Почему «один человек плюс ChatGPT» — неверная картинка

Solo AI-native контур — это не «один человек плюс ChatGPT». «Один человек плюс ChatGPT» — это режим, в котором оператор вручную формулирует каждый запрос в чате и сам переносит ответы в работу, оставаясь бутылочным горлом каждого действия. Эта формулировка предполагает, что человек сидит в чате, задаёт вопросы и копирует ответы. Такой режим упирается в потолок очень быстро: оператор становится бутылочным горлом для каждого действия, потому что ничего не происходит без его реплики. Контур, который мы описываем, устроен иначе. Это три конкретных сдвига в операционной модели, и каждый из них убирает оператора из части цикла.

Сдвиг первый: расписание вместо списка дел. В классической схеме человек держит to-do и сам решает, когда что делать. В AI-native контуре значительная часть работы запускается по расписанию, без участия оператора: research-контур собирает источники ночью, ops monitor проверяет здоровье систем

каждые два часа, утренний прогон готовит сводку дня. Список дел — это очередь, которую кто-то должен разбирать. Расписание — это конвейер, который работает сам и кладёт на стол готовые результаты. Разница не косметическая: список требует постоянного решения «что дальше», расписание это решение снимает.

Сдвиг второй: роли вместо мест. В команде из людей единица — это место (seat): человек, нанятый на позицию, с зарплатой и графиком. В агентном контуре единица — это роль: набор инструкций, памяти и инструментов, который выполняет функцию. Джек Дорси в письме о переходе Block к AI-first модели сформулировал ту же мысль: каждую функцию в компании можно усилить или заместить AI-native процессом, и мышление сдвигается от «сколько мест» к «какие роли» (Block). Роль не уходит в отпуск, не теряет контекст между задачами и масштабируется копированием. Пять ролей рядом с одним человеком — это не пять «сотрудников», это пять стабильных функций, которые он оркестрирует.

Сдвиг третий: артефакты вместо чат-треда. Это самый недооценённый из трёх. В чате знание живёт в треде: чтобы понять, что решено, нужно прокрутить переписку. Тред — плохая память: он линеен, не структурирован и теряется при переключении. В работающем контуре единица обмена — артефакт: файл с решением, отчёт, черновик, план. Агент не «отвечает в чате», а производит файл, который остаётся, версионируется и читается другими ролями. Эндрю Карпати в своём разборе LLM как новой операционной системы описывает память, инструменты и контекст как примитивы нового уровня абстракции — и именно артефакт, а не реплика в чате, становится носителем этой памяти (Andrej Karpathy). Когда все три сдвига на месте, оператор перестаёт быть участником каждого микро-действия и становится тем, кто запускает циклы и принимает их результаты.

Узкое горло — это циклы внимания

Здесь и проходит главная линия. Когда генерация дешёвая, а ролей много, узким местом становится не производство, а проверка. Каждый артефакт, прежде чем попасть в дело, проходит через одного человека — единственную точку, где есть стратегический контекст и право решать. Это и есть петля «задача → артефакт → ревью», и количество таких петель в день конечно.

Цифры здесь жёсткие. Исследование Глории Марк (Калифорнийский университет, Ирвайн) показало, что после прерывания человеку нужно в среднем около 23 минут, чтобы вернуться к исходной задаче. Для оператора, который держит три проекта и пять ролей, это означает: каждое переключение между контекстами стоит почти получаса восстановления. Поэтому контур, в котором агенты постоянно дёргают человека уточняющими вопросами, проигрывает контуру, где они производят законченные артефакты и приходят к оператору только на ревью. Параллелизм здесь когнитивный, а не технический: машина может вести десять задач одновременно, человек — одну в фокусе и несколь-

ко в фоне. Уиллисон описывает это честно: одна значимая петля в фокусе плюс несколько мелких в фоне, и быстрое истощение от переключений (Simon Willison).

Из этого следует контр-интуитивный вывод: добавить шестую агентную роль не всегда увеличивает пропускную способность контура. Если новая роль генерирует артефакты быстрее, чем человек успевает их проверять, она не ускоряет систему, а забивает очередь на ревью. Бутылочное горло просто сдвигается ближе к человеку. Поэтому правильная метрика зрелости такого контура — не число агентов и не объём генерации, а доля артефактов, которые проходят ревью и идут в дело, против тех, что копятся непроверенными.

Что отделяет работающий контур от витрины

Те, кто реально ведёт несколько проектов в одиночку, сходятся на нескольких признаках, которые отделяют рабочий контур от красивой витрины.

Первое — управление по исключениям. Оператор не читает весь поток. Системы спокойны по умолчанию и становятся заметны только там, где есть отклонение, риск или важный новый результат. Это прямой перенос логики mission-control в личный масштаб: если каждый сигнал красный и срочный, человек перестаёт воспринимать сигнал как значимый. Спокойный по умолчанию контур — это не эстетика, а способ не выжечь единственное узкое горло.

Второе — health-петли для знания. Артефактов накапливается много, и без регулярной проверки на устаревание контур начинает тихо врать: ссылки ломаются, отчёты ссылаются на отжившие факты, агент действует на основе протухшего контекста. За одну неделю мы видели, как research-контур производит десятки отчётов — и без сильной связки между research, решениями и действиями количество начинает обгонять пользу. Поэтому freshness-проверки, контроль битых ссылок и периодический пересмотр — это часть операционной системы, а не вторичная гигиена.

Третье — разделение потока и карты. Поток (входящие, уведомления, быстрые ответы) и карта (картина дня, приоритеты, состояние проектов) — это разные поверхности, и смешивать их вредно. Если карта живёт в том же месте, что и поток, она тонет в шуме. Управляющий слой для solo-оператора — это не BI-панель с сорока метриками, а навигационный слой поверх уже существующих поверхностей: что сейчас главное, где отклонение, какой следующий лучший переход вглубь.

Где этот контур ломается

Честность требует назвать границы. Solo AI-native контур выигрывает в скорости итерации и экономике — маржа на уровне 80–95% против отрицательной у команды до выхода на значимую выручку, как показывают разборы экономики solo+AI (Foundation Capital). Но он проигрывает команде в трёх конкретных местах.

Первое — глубина в специализированной области. Там, где нужна экспертиза, в которой модель галлюцинирует, один человек с агентами не заменяет специалиста. Второе — операции в режиме 24/7. Человек спит, и ночной инцидент упирается в единственную точку отказа: оператора. Ops monitor может разбудить, но не может принять решение вместо него. Третье — корпоративные продажи и комплаенс, где требуется человеческое присутствие в комнате. Sam Altman, говоря о компаниях из десяти человек с миллиардной оценкой, описывает именно прорыв в производительности, а не отмену всех ограничений (Every). Контур масштабирует производство и оркестрацию — он не масштабирует физическое присутствие и не закрывает дыру специализированной экспертизы.

Есть и более тонкий риск. Anthropic в разборе построения эффективных агентов предупреждает: добавление автономии увеличивает не только возможности, но и стоимость ошибок, и многие задачи лучше решать предсказуемыми процессами, а не агентами (Anthropic). Для solo-контура это означает: не каждую функцию стоит превращать в автономную роль. Там, где ошибка дорогая и тихая, лучше детерминированный сценарий с человеком на проверке, чем агент, которому дали свободу действовать.

Следствия для трёх типов читателей

Для solo-основателя проверочный вопрос один: где в вашем дне расписание уже заменило список дел? Если ответ «нигде» — вы ещё в режиме «человек плюс чат», и потолок близко. Начинать стоит не с того, чтобы добавить агентов, а с того, чтобы перевести хотя бы один повторяющийся процесс на расписание и научиться принимать его результат как артефакт, а не как реплику.

Для оператора, который внедряет AI в существующую команду, главное следствие — перестать думать местами. Вопрос не «кого из людей заменит агент», а «какие роли стоит выделить как стабильные функции с памятью и инструментами». И сразу за этим — где в команде узкое горло ревью, потому что именно туда упрётся весь прирост генерации.

Для руководителя, который строит AI-native org-дизайн, вывод жёстче. Метрика, которую стоит ставить во главу, — не headcount и не объём вывода, а пропускная способность по ревью: сколько решений с настоящим контекстом организация может принять в единицу времени. Это и есть настоящий потолок AI-native структуры. McKinsey описывает переход от org-chart к агентной сети как смену самой единицы организации — но единицей внимания всё равно остаётся человек, и его циклов конечное число.

Главное

- Solo AI-native контур — это не «один человек плюс ChatGPT», а три сдвига операционной модели: расписание вместо списка дел, роли вместо мест, артефакты вместо чат-треда.

- Узкое горло — не генерация кода или текста, а количество петель «задача → артефакт → ревью», которые один человек успевает закрыть за день; каждое переключение контекста стоит около 23 минут восстановления.
- Добавить ещё одну агентную роль не всегда ускоряет контур: если она генерирует быстрее, чем человек проверяет, она забивает очередь на ревью, а не разгружает её.
- Контур выигрывает в скорости и марже (80–95%), но проигрывает команде в специализированной глубине, операциях 24/7 и корпоративных продажах.
- Правильная метрика зрелости — не число агентов, а доля артефактов, прошедших ревью и пошедших в дело.

FAQ

Что такое solo AI-native контур и чем он отличается от «один человек плюс ChatGPT»?

Это операционная модель, где один оператор ведёт несколько проектов с помощью нескольких агентных ролей, работающих по расписанию и производящих готовые артефакты. Отличие от «человек плюс ChatGPT» в том, что оператор убран из части цикла: значительная часть работы запускается без его реплики, а он включается только на ревью. В чат-режиме человек — бутылочное горло для каждого действия; в контуре — только для проверки результатов.

Почему циклы внимания, а не генерация — это узкое горло?

Потому что генерация стала дешёвой и параллельной, а проверка осталась последовательной и привязанной к одному человеку с настоящим контекстом. Каждый артефакт проходит через одну точку принятия решения, и количество таких проверок в день конечно. Исследование Глории Марк даёт цену переключения — около 23 минут восстановления после прерывания, что делает беспорядочные переключения главным расходом внимания.

Когда этот паттерн неприменим?

Там, где нужна специализированная экспертиза, в которой модель галлюцинирует; в операциях 24/7, где человек как единственная точка отказа спит; и в корпоративных продажах с комплаенсом, где требуется присутствие человека в комнате. Контур масштабирует производство и оркестрацию, но не физическое присутствие и не глубину узкой области.

Сколько агентных ролей оптимально для одного оператора?

Универсального числа нет — потолок задаётся не количеством ролей, а пропускной способностью человека по ревью. Добавлять роль стоит только если человек успевает проверять её вывод; иначе она увеличивает очередь непроверенных артефактов, а не пропускную способность контура.

Как измерить, что контур работает, а не имитирует работу?

Главный показатель — доля артефактов, прошедших ревью и пошедших в дело, против накопленных непроверенными. Если очередь на ревью растёт быстрее, чем разбирается, контур производит видимость, а не результат, и узкое горло уже упёрлось в человека.