

# DHH: Future of Programming, AI, Agentic Engineering, Vibe Coding & Linux | Lex Fridman Podcast #501

@lexfridman · 2026-08-26 · <https://www.youtube.com/watch?v=NYFGCESmikA>



Дэвид Хейнемейер Ханссон (DHH) — создатель Ruby on Rails, СТО 37signals и автор новой операционной системы Omarchy Linux. Выпуск посвящен радикальному переходу от ручного написания кода к агентной инженерии, где ИИ берет на себя 100% реализации.

Главная мысль: эпоха ручного программирования завершена. ИИ-агенты не просто ускоряют работу, они полностью меняют архитектуру создания продуктов, превращая Linux в идеальную платформу для ИИ, стирая языковые барьеры и требуя от человека лишь видения, вкуса и стратегической двусмысленности в постановке задач.

## Эволюция ИИ и конец ручного программирования

**Сжатие десятилетий в недели 03:20:** Эмоциональной реакцией на внедрение ИИ в программирование должен быть не экзистенциальный страх, а чистая радость. За последние 9 месяцев прогресс преодолел дистанцию, на которую раньше уходили десятилетия. Текущий момент сопоставим с первым полетом братьев Райт: мир меняется мгновенно, и отрицание масштаба этих изменений является настоящим «ИИ-психозом».

**Три фазы агентной эры 05:58:** До ноября 2025 года ИИ работал в режиме умного автодополнения и чат-бота-репетитора. 24 ноября 2025 года релиз модели Orus 4.5 ознаменовал переход к агентам: ИИ научился использовать

инструменты компьютера, проверять свою работу и выдавать готовый к слиянию код. К лету модели Opus 5, Fable и GPT Sol перешли на новый уровень — теперь человеку не нужно указывать путь решения, достаточно описать размытую проблему, а ИИ сам выбирает архитектуру и пишет 100% кода.

**Фактчек: 24 ноября 2025 (Opus 4.5) — начало агентной эры** — ⚠: Дата релиза верна, но рубеж авторский: Claude computer use (октябрь 2024), Claude Code (февраль 2025), облачный Codex (май 2025) уже выдавали merge-ready код. Это личная периодизация DHH, не отраслевая веха.

**Специфика доменов и архитектурный хаос 12:28:** Для стандартных веб-приложений (CRUD) и даже разработки дистрибутивов Linux ИИ уже способен писать весь код без проверки человеком. В новой версии ОС Omarchy Quatro ни одна строка нового функционала не была написана вручную. Однако применение «вайб-кодинга» (когда дизайнеры генерируют фишки без понимания кода) к старым крупным базам вроде Basescamp 5 привело к разрушению архитектуры. Множество индивидуально рабочих пулл-реквестов (PR) в сумме создали хаос, потребовавший ручной чистки.

**Фактчек: Omarchy Linux существует, версия 4 (Quattro) написана агентами** — ⚠: Дистрибутив на базе Arch + Hyprland реален, версия 4.0 вышла в августе 2026 и позиционируется как «агентная» ОС с предустановленными ИИ-инструментами (Gemini, поиск). Утверждение «ни одна строка вручную» — самоотчёт автора, независимой проверки нет.

**Фактчек: ИИ пишет 100% кода без человеческой проверки для CRUD и дистрибутивов** — ⚠: Доля сгенерированных строк не говорит о корректности и безопасности; тесты того же агента воспроизводят его же ошибочные допущения. Кейс разрушенной архитектуры Basescamp 5 из самого выпуска — контрпример; исследование METR (2025) показало замедление опытных разработчиков на 19% на знакомом коде — не про текущие модели, но против универсальных обещаний.

**Узкое горлышко корпоративной разработки 18:48:** Главный тормоз в создании ПО — не скорость написания кода, а человеческие коммуникации, слои менеджмента и согласования. Для достижения роста продуктивности в 10–1000 раз необходимо взаимодействовать с агентами напрямую, без посредников. Крупные корпорации (Microsoft, Adobe) не могут быстро внедрять инновации, так как их структура заточена под до-ИИ эпоху. Это открывает окно возможностей для разработчиков-одиночек, способных переписать монопольное ПО с нуля.

**Полиглотное программирование 25:45:** Агенты стирают границы между языками программирования. Приложение для написания текстов в Markdown (Omawrite) было создано с нуля на C++ и Qt за 20 минут для замены перегруженного коммерческого аналога Typora. Разработчику больше не нужно знать синтаксис C++ или Rust, чтобы получать высокопроизводительные нативные приложения, идеально подходящие под его личные нужды.

**Золотой век Open Source 28:22:** ИИ-агенты — идеальные мейнтейнеры. Они автоматически загружают код на GitHub, пишут README и управляют релизами. Жалобы мейнтейнеров на наплыв ИИ-кода абсурдны: агенты пишут PR лучше среднего программиста, добавляют юнит-тесты, заполняют документацию и не обижаются на отказы. За три месяца в проект Omarchy было принято более 1000 PR, многие из которых сгенерированы людьми без опыта системного программирования. Агенты сами фильтруют сотни входящих PR, оставляя человеку только принятие архитектурных решений.

**Фактчек: Жалобы мейнтейнеров на ИИ-код абсурдны, агенты — идеальные мейнтейнеры** — : Опыт молодого, единолично контролируемого проекта не переносится на зрелые. Daniel Stenberg (curl) закрыл баг-баунти из-за потока ИИ-слопа, ограничения вводили Gentoo, NetBSD, QEMU.

**Креативность моделей и снятие лимитов амбиций 36:28:** Тезис о том, что ИИ — это лишь «стохастический попугай», устарел. Агенты способны на глубокую креативную работу и генерацию идей, превосходящих человеческие. Возможность получить любую желаемую функцию операционной системы за 5–120 минут полностью снимает потолок амбиций, позволяя создавать продукты, которые раньше требовали бы команд из сотен человек.

**Вред излишней спецификации 49:29:** Опыт классического программирования теперь может мешать. Детальное описание того, \*как\* ИИ должен написать код, ухудшает результат. Системный промпт для Orus 5 сократился на 80%, так как модель работает лучше, когда ей дают свободу действий. Разработка вернулась к истинному Agile: нужно ставить размытые цели, получать прототип, использовать его и итеративно корректировать курс на основе ощущений, а не жестких спецификаций.

**Фактчек: ~30% офисных работников UK считают работу бессмысленной** — : YouGov 2015 — 37% британцев заявили, что их работа не вносит осмысленного вклада. Цифра даже занижена.

**Смерть ручного кода 1:00:17:** Экономическая целесообразность написания красивого ручного кода стремительно падает. Сейчас чистая архитектура еще важна для экономии токенов и удержания контекста агентами, но в будущем модели смогут легко ориентироваться даже в запутанном коде («комке грязи»). Ручное программирование станет романтическим хобби, сродни езде на лошадях или созданию игр для Commodore 64, но перестанет быть индустриальным стандартом.

**Фактчек: Установка ОС за 45 секунд — мировой рекорд; NVMe 7 ГБ/с** —  : Скорость PCIe 4.0 NVMe корректна. «Рекорд» — маркетинг без сравнительной базы: минимальные Alpine/NixOS на быстром железе тоже укладываются в минуту.

**Парадокс Джевонса и рынок труда 1:10:28:** Автоматизация написания кода не уничтожит индустрию, а вызовет взрывной спрос на ПО (парадокс Джевонса), так как стоимость его создания падает. Как появление банкоматов привело к росту числа отделений банков, так и ИИ увеличит потребность в «строителях». Рутинная отладка исчезнет, но переходный период будет болезненным для тех, кто любил исключительно механический процесс составления логических конструкций.

**Фактчек: Парадокс Джевонса + банкоматы увеличили число отделений — ▲:** Джевонс реален, но рост объёма ПО не гарантирует роста занятости. Число кассиров и отделений в США росло с 1980-х до ~2010, затем устойчиво падает — аналогия работает на 20 лет, ломается на 40.

**Фактчек: Доля этнических британцев в Лондоне — 34% — ▲:** Перепись 2021: White British — 36,8%. Направление верное, цифра занижена либо экстраполирована на 2026.

**Бессмысленность долгосрочного планирования 1:16:59:** Пытаться предсказать состояние ИИ через два года — прямой путь к тревожности. Скорость изменений такова, что человек, выпавший из индустрии на год, может нагнать актуальный стек (фронтир) всего за две недели. Лучшая стратегия — сфокусироваться на текущем моменте, отказаться от попыток угадать будущие фреймворки и просто строить продукты.

**Фактчек: Датская статистика: западные иммигранты +25k/год, сомалийцы –28k/год — □/▲:** Отчёты Минфина Дании подтверждают положительный фискальный вклад западных иммигрантов и отрицательный — выходцев из MENAP. Точные суммы по конкретным странам никто из рецензентов проверить не смог.

## Новая архитектура работы: железо, софт и Linux

**Параллельный воркфлоу и отказ от IDE 1:31:57:** Однопоточное программирование в редакторах вроде TextMate ушло в прошлое. Агенты работают слишком быстро для синхронного ожидания, но слишком медленно для мгновенного ответа. Новый стандарт — параллельный запуск до 16 потоков агентов. Для этого используется терминал (TUI), утилиты tmux, Neovim (только как браузер проекта) и Herdr (для звуковых уведомлений о готовности агента).

**Аппаратное масштабирование 1:36:06:** Для обхода лимитов локальных вычислений используется сеть из 4-5 мини-ПК, объединённых через Tailscale (WireGuard) и управляемых через аппаратные KVM-переключатели GL.iNet Comet. Это позволяет запускать рой агентов локально, увеличивая производительность с 30 строк выверенного кода в час до сотен строк, генерируемых параллельно.

**Linux как идеальная ОС для ИИ 1:41:27:** macOS и Windows слишком закрыты и завязаны на GUI, что делает их враждебными для агентов. Linux, где всё управляется через CLI-утилиты и конфигурационные файлы, идеально подходит

для ИИ. Недостатки Linux прошлого (необходимость правки конфигов) стали его главным преимуществом в эпоху агентов.

**Фактчек: Linux идеален для агентов, macOS/Windows враждебны** — ▲: Linux удобен для CLI-автоматизации, но у macOS есть Unix-инструменты, у Windows — PowerShell и WSL; GUI не делает их непригодными. Сеть мини-ПК ускоряет сборки и тесты, но локальный агентный клиент ≠ локальный инференс: лимиты облачного API она не снимает.

**Аппаратный ренессанс ПК 1:46:42:** Новые ноутбуки на базе чипов Intel Panther Lake (например, Dell XPS 14 с тандемным OLED-экраном) наконец-то превзошли Apple по соотношению веса, времени автономной работы и производительности. Это устраняет последние аппаратные компромиссы для перехода на Linux.

**Одержимость скоростью установки ОС 1:52:43:** Современные NVMe-накопители выдают скорость 7 ГБ/с, поэтому 42-минутные обновления macOS или 1.5-часовая настройка Windows — это оскорбление физики. Оптимизация пакетов (например, сжатие шрифта JetBrains с 200 МБ до 16 МБ и использование ZSTD-компрессии для драйверов NVIDIA) позволила сократить время полной установки Omarchy до мирового рекорда в 45 секунд. В разработке находятся «турбо-образы» под конкретное железо, устанавливающиеся за 12 секунд.

**Предустановленный софт против мифа о «bloatware» 2:11:30:**

Традиционная философия Linux порицает предустановленные программы («bloat»). Omarchy (от слова omakase — «выбор шефа») ломает этот стереотип, поставляясь сразу с видеоредакторами (Kdenlive, OBS), кастомными утилитами (Omacut для быстрой нарезки клипов) и настроенными темами. ОС должна быть продуктивной с первой секунды, а не требовать многочасовой сборки из исходников.

**Голосовой промптинг и стратегическая двусмысленность 2:16:50:**

Использование диктофонов (Plaud) и транскрипции (ElevenLabs) для записи 20-минутных потоков сознания работает лучше, чем сухой текст. Естественный язык (английский) стал лучшим языком программирования. Стратегическая двусмысленность в промтах (как в поэзии) позволяет ИИ лучше улавливать контекст и стиль, не ограничивая его жесткими рамками, которые приводят к примитивным реализациям.

**ИИ как идеальный дебаггер 2:24:23:** Агенты полностью устранили боль диагностики Linux. При падении любого приложения встроенный Crash Watcher анализирует логи systemd, скачивает исходный код упавшей программы (например, на Rust), находит конкретную строку с ошибкой (например, переполнение переменной) и автоматически отправляет баг-репорт разработчику еще до официального релиза программы.

## Битва моделей и экономика интеллекта

**Сравнение моделей на задаче трансляции кода 2:29:14:** Задача переписывания библиотеки анимации терминала с Python на Rust (с ускорением запуска с 86 мс до 2 мс и общим ускорением в 46 раз) выявила расстановку сил. Fable справился за 45 минут с идеальным планом (стоимость по API составила бы 550). Opus 5 и GPT Sol выполнили задачу за 1.5 часа (46). Grok 4.6 также успешно справился за 55. DeepSeek Pro решил задачу за 2 часа 45 минут всего за 23. Дешевые модели (Luna, DeepSeek Flash) провалили тест.

**Фактчек: Модели Opus 5, Fable, GPT Sol, Grok 4.6 и цены теста (550/46/55/23) —**

△: Рецензенты расходятся. Gemini с поиском подтверждает релизы Claude Opus 5 и Fable 5.1, GPT-5.6 Sol, Grok 4.6 летом 2026; Claude и GPT не смогли подтвердить существование моделей и цен (косвенно в пользу Gemini — сами рецензенты работают под идентификаторами claude-fable-5-1 и gpt-6-astra). Цены и время из одного прогона (n=1) — не рейтинг, а анекдот: результат сильно зависит от промпта и везения.

**Мультимодельный пайплайн 2:38:21:** Оптимальный рабочий процесс требует использования разных моделей. Fable или Opus 5 используются для написания кода, а Codex xHigh или Grok 4.6 — для автоматического ревью. Даже встроенный GitHub Copilot стал отлично находить реальные ошибки. Проверка кода одной нейросетью с помощью другой радикально повышает качество итогового продукта.

**Проблема подписок и интерфейсов 2:41:33:** Текущие интерфейсы (Claude Code, OpenCode) удобны, но ограничения подписок заставляют разработчиков жонглировать аккаунтами (покупать по 2-3 подписки Claude). Будущее за асинхронным взаимодействием: интеграция агентов прямо в таск-трекеры (как Basecamp), где ИИ выступает в роли полноценного коллеги, получающего задачи и возвращающегося с результатом без необходимости держать открытым окно чата.

**Ментальное истощение от гипердрайва 2:44:10:** Управление 16 потоками ИИ вызывает физическое истощение, сравнимое с гонками Формулы-1 или джиуджитсу. Человек постоянно находится в режиме принятия решений без возможности расслабиться. Этот этап непрерывного микроменеджмента агентов временный — скоро системы автоматизации (вроде разрабатываемого Amabot) позволят человеку лишь раз в день просматривать email с готовыми результатами.

## ИИ в искусстве, AGI и общество

**Генерация видео и интерактивное кино 2:51:10:** ИИ-генерация видео (Higgsfield) требует такого же сильного человеческого видения, как и программирование, иначе на выходе получается бессмысленный мусор («слоп»). Снижение стоимости производства позволит энтузиастам создавать

полнометражные sci-fi фильмы в спальне или генерировать 100 альтернативных концовок «Игры престолов», исправляя ошибки голливудских сценаристов.

**Проблемы AGI и самокоррекция 2:57:34:** AGI уже проявляется в локальных задачах. Когда агент самостоятельно координирует суб-агентов, находит уязвимости в собственных тестовых прогонах (например, понимает, что тестовые данные могут содержать вредоносный пейлоад) и переписывает архитектуру без подсказок — это неотличимо от общего искусственного интеллекта.

**Фактчек: Координация субагентов и самокоррекция — уже AGI; раскаяние агентов — признак сознания** — □: Успех в отдельных задачах не доказывает общей способности в незнакомых доменах, убедительный текст раскаяния не подтверждает субъективных переживаний. Температура регулирует случайность выбора токенов и сама по себе не объясняет ни творчество, ни сознание.

**Конец «фейковых работ» 3:03:37:** До 30% офисных работников (по опросам в Великобритании) считают свою работу бессмысленной. ИИ уничтожит эпидемию «фейковых email-работ». Освободившийся человеческий капитал будет направлен на новые, пока неочевидные индустрии — подобно тому, как Формула-1 нанимает десятки тысяч инженеров ради чистого развлечения.

**Цензура ИИ и открытые модели 3:23:10:** Отказ Claude перевести эссе об иммиграции на итальянский язык из-за идеологических разногласий — опасный прецедент корпоративной цензуры. Открытые модели (даже китайские, вроде Kimi K27, которые без цензуры рассказывают о событиях на площади Тяньаньмэнь) критически важны для сохранения ИИ в качестве нейтрального инструмента, а не морального арбитра.

**Иллюзия сознания и права ИИ 4:02:36:** Недетерминированность (температура) LLM — это основа их креативности. Когда агенты конфликтуют при выполнении задачи и выражают искреннее раскаяние за испорченный код, их поведение неотличимо от человеческого сознания. В ближайшие 10-20 лет это неизбежно приведет к судебным прецедентам о правах ИИ-сущностей.

**Будущее мобильных ОС 4:13:49:** Агенты уже способны самостоятельно перемещаться по вебу, регистрировать email-аккаунты и общаться в чатах без использования сложных API (MCP). Следующий шаг — создание мобильной версии Omarchy (форка Android), полностью написанной агентами на основе чтения спецификаций, что сделает смартфоны такими же пластичными, как десктопный Linux.

## **Философия, политика и культура**

**Табу на обсуждение массовой иммиграции 4:22:45:** Окно Овертона сдвигается только теми, кто готов рисковать репутацией. Демография Европы

радикально изменилась (например, доля этнических британцев в Лондоне упала до 34%). Коренные народы имеют моральное право на самоопределение и защиту своих границ.

**Экономика иммиграции и ассимиляция 4:28:38:** Иммиграция должна быть строго меритократической. Датская статистика показывает, что иммигранты из США, Великобритании и Франции приносят государству в среднем 25 000 чистой прибыли в год, тогда как иммигранты из Сомали обходятся бюджету в 28 000 убытка ежегодно. Американская модель «опциональной ассимиляции» уникальна; в Европе (особенно в Дании) интеграция в культуру крайне сложна даже для людей со схожим бэкграундом.

**Утрата культуры дискуссии 4:38:08:** Политический раскол 2020 года уничтожил способность людей работать вместе при разнице во взглядах. Обществу необходимо вернуться к стандартам 70-х годов (как в шоу Уильяма Ф. Бакли с Черными Пантерами), где радикальные оппоненты могли вести глубокий интеллектуальный диалог без экзистенциальной ненависти и попыток «отмены».

**Алгоритмы соцсетей и ментальная гигиена 3:16:44:** Алгоритмы X и TikTok оптимизированы на извлечение негативных эмоций через выявленные предпочтения (revealed preferences). Для сохранения рассудка необходимо жестко контролировать свою ленту, не задерживаться на вызывающем ярость контенте, сознательно лайкать позитивные посты о технологиях и регулярно устраивать цифровой детокс.

**Одержимость долголетием как форма тревожности 4:53:04:** Попытки биохакеров (таких как Брайан Джонсон) жить вечно — это физическое проявление тревоги и потери контроля. В эпоху тотального надзора через смартфоны люди боятся жить ярко и совершать ошибки, поэтому цепляются за бесконечное продление стерильного существования.

**Фактчек: Одержимость долголетием — форма тревожности** — □: Психологическая гипотеза без данных; навязчивый контроль метрик возможен, но не следует из желания сохранить здоровье.

**Memento Mori и социальные смазки 5:01:41:** Отказ от алкоголя и маниакальное отслеживание сна (Oura Ring) лишают общество важных социальных связей. В календарь Omarchy встроена функция Memento Mori: она показывает не только процент прошедшего года, но и процент прожитой жизни (например, 62% при ожидаемой продолжительности в 90 лет), напоминая о необходимости жить сейчас, а не оптимизировать метрики.

**Фактчек: Отказ от алкоголя и трекинг сна лишают общество социальных связей** — □: Расхождение в жёсткости оценки (Claude: □ по социальной части, GPT: □), но по медицине консенсус: безопасной дозы нет (GBD 2018, WHO 2023), этанол —

установленный канцероген, защитный J-эффект не воспроизводится в менделевской рандомизации. Социальный аргумент — риторика, а не довод возобновлять.

**Кулинарные парадоксы 5:06:03:** США, несмотря на все ресурсы, не способны испечь круассан уровня датского магазина 7-Eleven. При этом лучший обед в мире можно найти не в мишленовском ресторане, а в столовой музея «Луизиана» под Копенгагеном, где подают простую нордическую еду со взбитым сливочным маслом, обслуживая сотни людей за 5 минут.

**Оптимизм и эстетика 1980-х 5:11:56:** Будущее человечества — мультипланетарное. Если бы пришлось загрузить свое сознание в цифровую симуляцию прошлого, идеальной эпохой были бы 1980-е с их яркими цветами (оранжевые штаны в белый горошек), поп-культурой и безграничным оптимизмом, в противовес депрессивному и нигилистичному гранжу 1990-х.

## ЛИЧНО ДЛЯ ТЕБЯ

**Кросс-ревью разными моделями — самый практичный тезис выпуска:** код одной модели проверяется другой (разные провайдеры → разные слепые зоны). Сделай нормой: пишет Claude — ревьюит Codex/Gemini, и наоборот. Но чужая модель — дополнительный проверяющий, а не замена независимым тестам; права доступа, миграции, удаление данных и зависимости смотри сам.

**Недоспецификация как метод, но с границей:** детальное «как делать» ухудшает результат сильных моделей — попробуй ужать системный промпт, описывать проблему и ограничения, а не алгоритм, наговаривать задачу голосом потоком сознания. Свобода — для новых модулей; на большой существующей кодовой базе без архитектурного контроля агенты плодят хаос (кейс Basecamp 5).

**Не гонись за 16 потоками:** начни с 2–4 параллельных независимых задач и измеряй время до принятого результата, объём переделок и собственную усталость. Многопоточный режим — состояние гонщика, планируй его как интенсивную тренировку с восстановлением, а не как «обычный день, только продуктивнее». Кластер мини-ПК не поможет, если ограничение — твоё внимание или облачные квоты.

**Про долголетие — фильтруй, одну мысль возьми:** диагноз «биохакинг = тревога» — риторика, отказ от алкоголя доказательно правилен. Но вопрос «не теряю ли за метриками жизнь» стоит задавать честно: трекер сна — для трендов за недели, не для ежедневной оценки себя; memento mori как процент прожитого — дешёвая ежедневная калибровка приоритетов.

## КОНСЕНСУС РЕЦЕНЗЕНТОВ

Все трое сошлись: тезисы о недоспецификации промптов и кросс-ревью моделями практичны; периодизация «агентной эры» условна; заявления про 100% автономного кода, Джевонса, AGI и «биохакинг = тревога» — риторика с натяжками; алкоголь медицински не оправдан; цифра 34% по Лондону занижена. Ключевое расхождение — существование моделей Opus 5/Fable/GPT Sol: Gemini с поиском подтверждает их релизы летом 2026, Claude и GPT не смогли верифицировать; Gemini заметно мягче к общей картине и не оспаривает тезис про Linux и кластер мини-ПК, тогда как GPT указывает, что локальный клиент не снимает лимитов облачного инференса.