

● ОТКРЫТАЯ ОНЛАЙН-ВСТРЕЧА

Лучшие практики вайбкодинга

Как я на самом деле работаю с ИИ-агентами каждый день

Спикер — **Александр Шкарупа**, основатель PlusvibeAPI.ru

Сначала я, потом вы

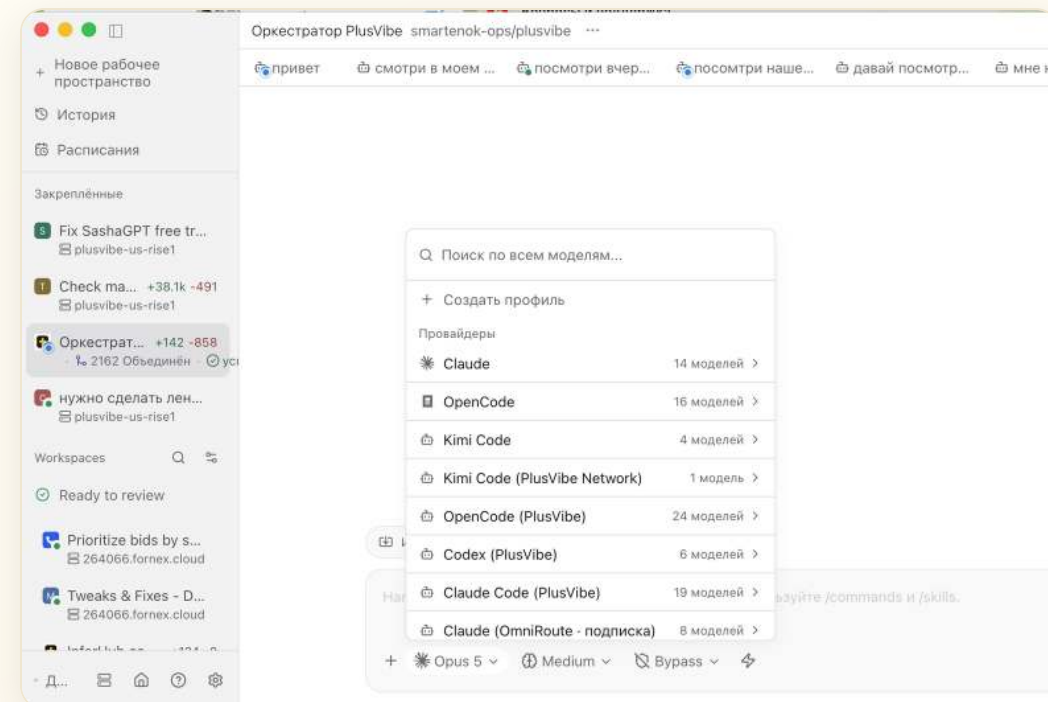
- **15 минут — я.** Мой путь и рабочая схема.
- **Вопросы и ответы.** Детали и спорные места.
- **Общая дискуссия.** Ваши модели, инструменты, опыт.

Я перепробовал разные модели и инструменты. Покажу, на чём остановился и почему.

Paseo: все агенты в одном окне

- Один экран на всех: **Claude**, **Codex**, **OpenCode**, **Kimi**.
- Видно, кто занят, кто закончил, кто ждёт ответа.
- Модель новому агенту выбираю из общего списка.
- Работает с телефона.

paseo.sh — то, чем пользуюсь **onorca.dev** — альтернатива



СВЯЗКИ, КОТОРЫМИ ПОЛЬЗУЮСЬ

ChatGPT
+ Codex

Opus 5
+ Claude Code

Китайские модели
+ OpenCode
или DeepSeek harness, или Oh My Pi — omp.sh

ГДЕ ЖИВУТ АГЕНТЫ

VPS вместо ноутбука



ЧТО МЕШАЛО НА НОУТБУКЕ

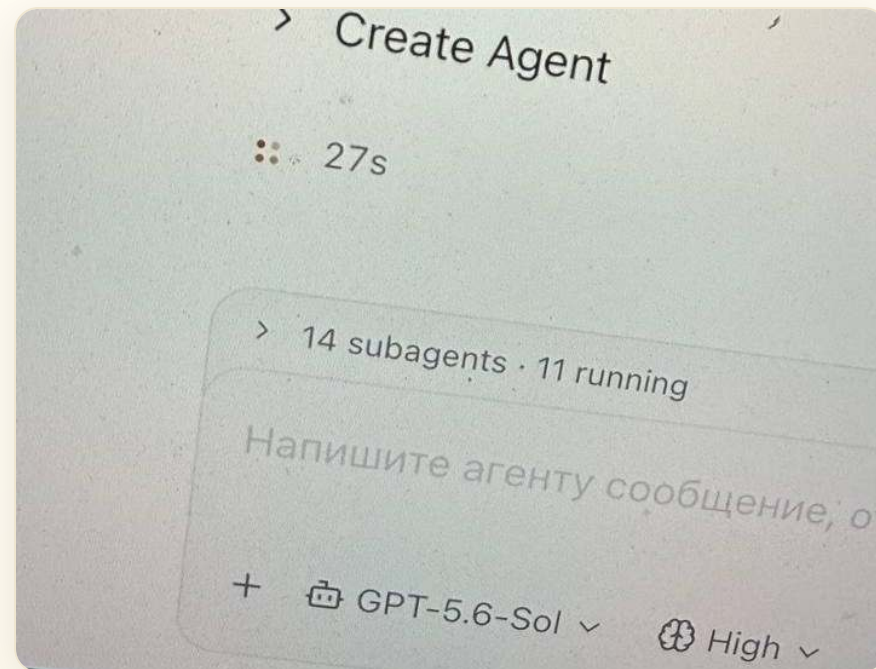
- Память кончается первой: сборка и агент рядом не помещаются.
- Одна задача за раз — второй агент затирает работу первого.
- Закрыл крышку — всё встало.
- Некому убить зависшую сборку и разгрести диск.

ЧТО ДАЁТ СЕРВЕР

- Проекты живут на этой же машине — **26 запущенных сервисов**.
- У каждого агента **своя копия проекта** и своя ветка. Сейчас 13 параллельно.
- Агенты работают, когда я отошёл или сплю.

Один ставит задачи, остальные пишут код

- Оркестратор **сам код не пишет**. Ставит задачу и читает готовые правки.
- Исполнитель получает ровно четыре вещи: **задачу, файлы, признаки готовности, границы**.
- **Один уровень вложенности**. Исполнитель своих исполнителей не заводит.
- Исполнитель **ничего не выкладывает на боевой сайт**, не сливает работу в основную ветку и не видит паролей.
- Результат проверяет сам оркестратор. Отдельного проверяющего на обычную задачу нет.



Так это выглядит в интерфейсе: **14 исполнителей заведено, 11 работают прямо сейчас**. Модель и глубину размышления выбираю здесь же.

Модель под задачу, а не одна на всё

ОРКЕСТРАТОР И СЛОЖНЫЕ ЗАДАЧИ

claude-opus-5

Деньги, цены, база, безопасность, распределение запросов между поставщиками. Всё, где ошибка дорогая.

Так же подходят: **gpt-5.6-sol**, **Kimi K3**, **glm-5.3**.

ОБЫЧНОЕ

gpt-5.6-luna

Новые функции, обновление веток, починка тестов, переделка кода. Основной объём работы.

Для задач посложнее — **gpt-5.6-terra** или **deepseek-v4-pro**.

МЕХАНИЧЕСКОЕ

deepseek-v4-flash, glm-5.3-flash

Переименования, переносы документации, скрипты для замеров, однотипные правки.

45%

137 871 запрос

36%

109 488 запросов

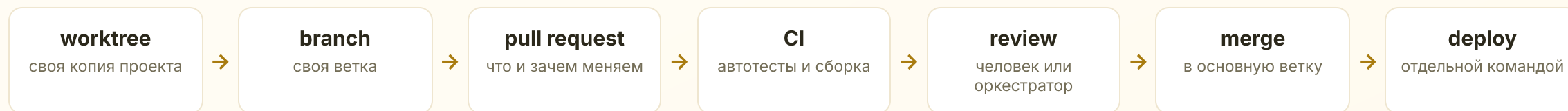
19%

58 424 запроса

Решения принимает дорогая модель, исполняют дешёвые. **Все они идут через мой же API:** один ключ, общий счёт, поставщика меняю, не трогая агента.

КАК АГЕНТЫ ДОГОВАРИВАЮТСЯ МЕЖДУ СОБОЙ

GitHub: от worktree до deploy



- **Pull request** — единственная точка, где изменение ещё можно остановить. Поэтому в основную ветку агент не пишет.
- Красный CI — **ошибка в этом pull request**, а не в тестах. Тест не удаляется, pull request не мержится.
- **Merge** — это ещё не **deploy**. Выкладка идёт отдельной командой, и агент её не запускает.
- CI сначала гонял в **GitHub Actions** за \$4 в месяц. Упёрся в лимит **3000 минут** — перенёс сборку на свой сервер.

Что получилось за месяц

340 461

обращение к моделям
11 300 в сутки, в среднем

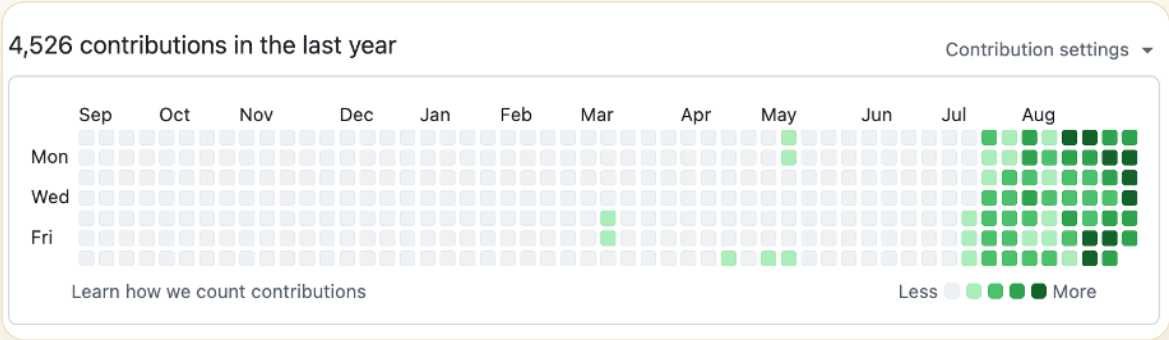
1 441

pull request принят
1 982 запуска агентов

1 360

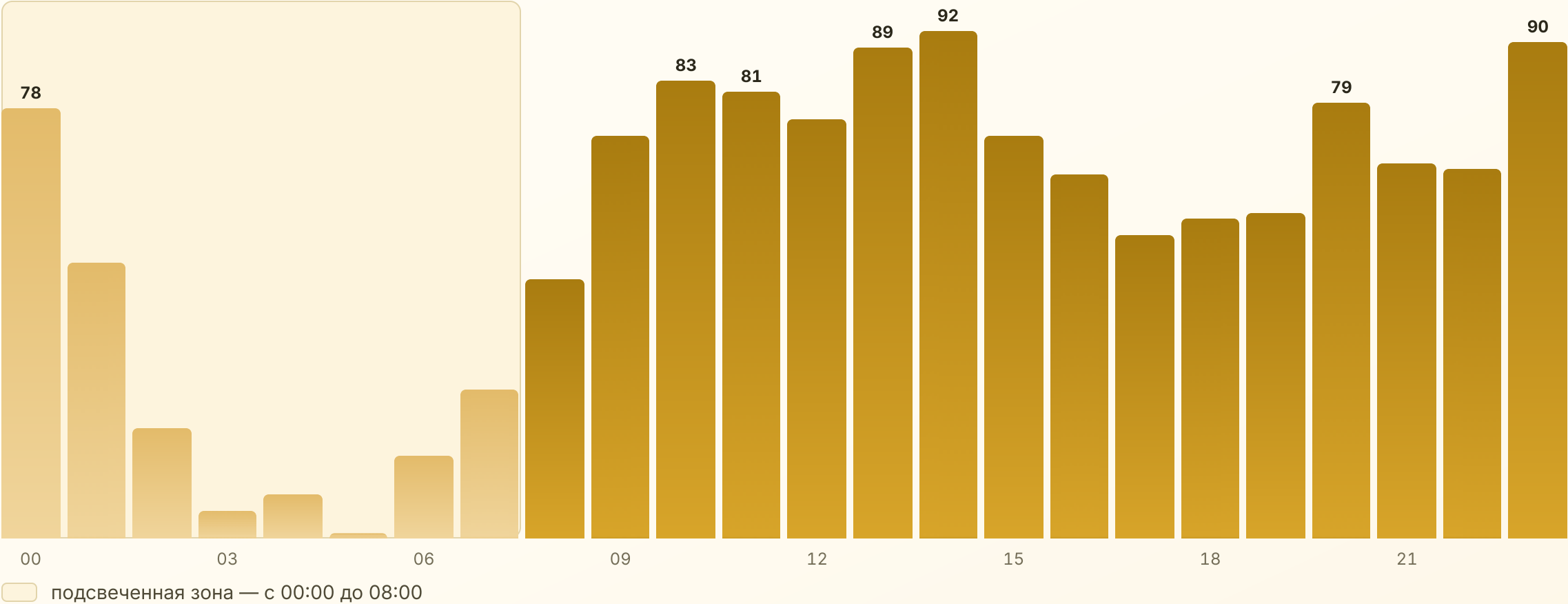
КОММИТОВ В ОСНОВНОЙ ВЕТКЕ
89 живых веток

МОДЕЛЬ	ЗАПРОСОВ
claude-opus-5	137 871
gpt-5.6-luna	109 488
deepseek-v4-flash	38 342
deepseek-v4-pro	17 602
gpt-5.6-terra	15 868
gpt-5.6-sol	9 058
остальные модели	12 232
Всего	340 461



За год **4 526 коммитов**. Плотная заливка идёт **с середины июля**: именно тогда я начал работать так, как только что рассказал.

Ночью работа не останавливается



204 коммита из 1 360 прились на ночь. Оркестратор раз в ~2 часа передаёт смену следующему через файл, цели записаны заранее, подстраховывает простой скрипт по расписанию.

РАСПИСАНИЕ

Часть работы идёт вообще без меня

УБОРКА

Чистит диск, гасит зависшие сборки, усыпляет агентов, которые простаивают.

СТОРОЖА

Проверяют, что сервисы живы, и поднимают упавшее раньше, чем я это замечу.

БЭКАПЫ

Ночные копии базы — и регулярная проверка, что копия действительно разворачивается.

55 строк расписания плюс 6 сторожевых скриптов у самой панели агентов. Пишутся они не заранее, а после того, как что-то один раз сломалось.

Расписание — не про то, чтобы агент работал ночью. Оно про то, чтобы к утру ничего не развалилось.

Откуда агент знает правила проекта

ГРУЗИТСЯ ВСЕГДА · 60 КБ

AGENTS.md

манифест проекта: правила, границы, запреты

CLAUDE.md

одна строка: **@./AGENTS.md** — чтобы Claude и Codex читали один файл и не разъезжались

MEMORY.md

оглавление памяти: заголовки, без текста

ЛЕЖИТ РЯДОМ · 7,4 МБ

docs/agent/*.md

5 разборов: цены, деплой, провайдеры, таймауты

docs/runbooks/*.md

72 инструкции под конкретную операцию

memory/*.md

296 фактов: почему так решили и где обожглись

Правое агент не читает заранее — открывает сам, когда упёрся в вопрос. Контекст расходник: чем больше свалить в начале, тем меньше останется на задачу.

Чего в файле нет — того агент не знает. Он не догадается, он выдумает. Поэтому каждая полочка заканчивается новой строкой в инструкции, а не разговором.

ЧТО ВАЖНО ЗНАТЬ ЗАРАНЕЕ

Что у меня ломалось и с чего начать

ЧТО ЛОМАЛОСЬ

- Агент отчитывается о том, чего не делал. **Его отчёт — не доказательство.**
- В долгой работе начинает выдумывать задачи от вашего имени.
- Сервер один раз завис из-за нехватки памяти на 53 минуты.
- Два агента берут одну ветку и затирают работу друг друга.

С ЧЕГО НАЧАТЬ

- Сервер — хватит небольшого.
- Отдельная копия проекта на каждую задачу.
- Ничего не мержить в основную ветку без pull request.
- Одна дорогая модель на решения, одна дешёвая на исполнение.
- Ночные смены — потом, не на старте.

• ДАЛЬШЕ — ВАША ЧАСТЬ ВСТРЕЧИ

Вопросы и обсуждение

Спасибо. Спрашивайте про любую часть схемы, расскажите, чем пользуетесь сами.

PlusvibeAPI.ru