

Что и зачем мерить в цифровом двойнике команды

Восемь страниц о том, какие цифры говорят правду о потоке работы, как их читать и чего от них не требовать.

Цифровой двойник команды — это не дашборд и не таблица KPI. Это память о том, что происходило: каждый переход задачи по статусам, каждая запись о событии в команде, каждый списанный час — отдельное событие с датой, автором и причиной. Ничего не удаляется и не правится задним числом. Все метрики ниже — не числа, которые кто-то ввёл, а проекции над этими событиями: любую цифру можно раскрутить назад до строки выгрузки из трекера.

Поток задач

переходы статусов из трекера: кто, что, когда и куда двигал. Основа всех метрик потока.

Журнал команды

словесные события: отпуск, ретро, инцидент, смена процесса. Объясняет, почему метрики изменились.

Списания времени

часы на подзадачи. Показывают, где работа была, а где задача просто ждала.

Главное правило. Мы измеряем поток работы, а не людей. Lead time, очереди, возвраты и WIP — свойства системы: процесса, передач между этапами, размера задач. Если цифра используется, чтобы оценить человека, она перестаёт быть честной: люди начинают оптимизировать цифру, а не работу.

Пять вопросов, на которые должны отвечать цифры

1. Что происходит?

Сколько дней идёт задача, сколько закрываем в неделю, сколько открыто одновременно. Lead time, пропускная способность, WIP.

2. Где теряем время?

В каком статусе задачи стоят, а не делаются. Очереди почти всегда больше работы. Время в статусе, эффективность потока, CFD, старение.

3. Почему?

Что происходило в команде, когда метрики изменились. Событие рядом с изломом — гипотеза, не доказательство. Журнал команды, импакт «до → после», изломы.

4. Что будет?

Когда закроется эпик, если команда работает как сейчас. Не одна дата, а разброс. Прогноз Монте-Карло, what-if на двойнике.

5. Откуда знаем?

Каждая цифра должна раскручиваться до событий, из которых посчитана. Кнопка «f» у плитки, граф событий, запрос к базе. Цифра без провенанса на решения не влияет.

Карта метрик: что, зачем, как читать

Третья колонка — главное. Метрика без вопроса, на который она отвечает, превращается в украшение дашборда.

Метрика	Что это	Зачем и как читать	Где в Vanilitik
Lead time p50 / p85	Дни от создания задачи до закрытия. Всё время, включая ожидание. То, что видит заказчик.	Обещание сроков: «85% задач закрываем за N дней». p85 честнее среднего: его не тянут вверх редкие долгие задачи. Рост p85 при том же p50 — появился хвост застрявших.	Отчёты → KPI; динамика по периодам
Cycle time	Дни от первого рабочего статуса до закрытия. То, что видит команда.	Разница lead – cycle = сколько задача ждала до начала работы. Створ «откуда / докуда» позволяет мерить любой отрезок: до разработки, разработка, после.	KPI → переключатель Lead / Cycle, створ
Пропускная способность	Сколько задач закрыто в неделю.	Ритм поставки. Пила раз в две недели — релизный поезд; ровная линия — непрерывная поставка. Сравнить команды нельзя (разный размер задач), сравнить команду с собой по периодам — можно.	KPI; Throughput по неделям
WIP	Сколько задач открыто одновременно (и на человека).	Закон Литтла: $lead \approx WIP \div \text{пропускная}$. Больше 3–4 открытых задач на человека — переключения съедают время, каждая задача идёт дольше. Самый дешёвый рычаг: ограничить WIP.	KPI «В работе»; Datalog «открытые по исполнителям»
Эффективность потока	$\text{Работа} \div (\text{работа} + \text{очереди})$, по пути завершённых задач.	Меньше 40% — команда в основном ждёт, ускорять надо передачи между этапами, а не людей. Самая важная цифра для разговора «нам нужно больше разработчиков».	KPI; «Из чего состоит lead: работа vs очереди»
Время в статусе	Среднее и медианное время, которое задачи проводят в каждом статусе.	Столбик очереди (wait) выше остальных — узкое место. Называть узкое место надо статусом: «задачи ждут в “Готова к внедрению” 5 дней», а не «медленно внедряем».	Среднее время в статусе; минус стоянки
CFD	Накопительная диаграмма: сколько задач в каждом статусе день за днём.	Расширяющаяся полоса — очередь растёт; параллельные полосы — поток стабилен. Линейка на двух датах даёт WIP и приближённый lead. Сузьте окно дат до подозрительных недель.	Накопительная диаграмма потока (CFD)
Старение WIP	Открытые задачи по времени в текущем статусе.	Топ застрявших прямо сейчас. Если все в одном статусе — это и есть ответ. Если застрявших нет — команда здорова сейчас, а проблема была раньше: смотрите CFD.	Застрявшие задачи (Aging WIP)
Возвраты	Переходы назад: из тестирования в разработку.	Переделка. Фон 5–15% нормален; концентрация на компоненте или после определённого события — сигнал о качестве постановки или ревью. Не про «виноватого»: у одного человека могут быть самые сложные задачи.	Матрица переходов; Datalog «возвраты по исполнителям»

Минуты этапы	Закрытые задачи без события тестирования (или другого обязательного этапа).	Что уходит в прод без проверки. Единицы — норма (документация, конфиги); много у одной команды — риск, смотрите её инциденты следом.	Datalog «закрывают, минуя тестирование»
Списано vs календарь	Часы на задачу против дней, которые она шла. Плотность = часы ÷ (дни в работе × 8).	Точки под 2 ч/день — задача в основном ждала; над 8 ч/день — параллельная работа или переработки. Два горба в распределении — два режима учёта: честный и «для галочки».	Списанное время и эффективность
Журнал и импакт	События команды и метрики за окно до и после каждого.	Связывает цифры с жизнью команды. Окно 14 дней ловит резкие изломы (отпуск, заморозка), 28 — медленные (ретро, новый процесс). Событие без эффекта — тоже находка.	Анализ событий и трендов
Изломы трендов	Недели, после которых метрика устойчиво сменила уровень.	Излом рядом с событием журнала — гипотеза для расследования. Излом без объяснения — повод дописать журнал: значит, что-то произошло, а команда этого не записала.	Изломы трендов; «рядом в журнале»
Эпики и прогноз	Длительность эпиков и Монте-Карло по историческому throughput.	Не «когда», а «с какой вероятностью к какой дате»: P50 / P85 / P95. What-if показывает, что было бы без релизного поезда или с половиной очередей — рычаг в днях, а не в лозунгах.	Эпики; Прогноз Монте-Карло; What-if

CFD: как читать накопительную диаграмму потока

CFD (cumulative flow diagram) — главный экран для поиска очередей. По горизонтали дни, по вертикали число задач. Каждый день считается, сколько задач **уже дошли** до каждого статуса: сколько создано, сколько дошло до разработки, до тестирования, до закрытия. Эти счётчики только растут, поэтому линии не падают, а полосы между ними показывают, сколько задач **сейчас находится** в каждом статусе.

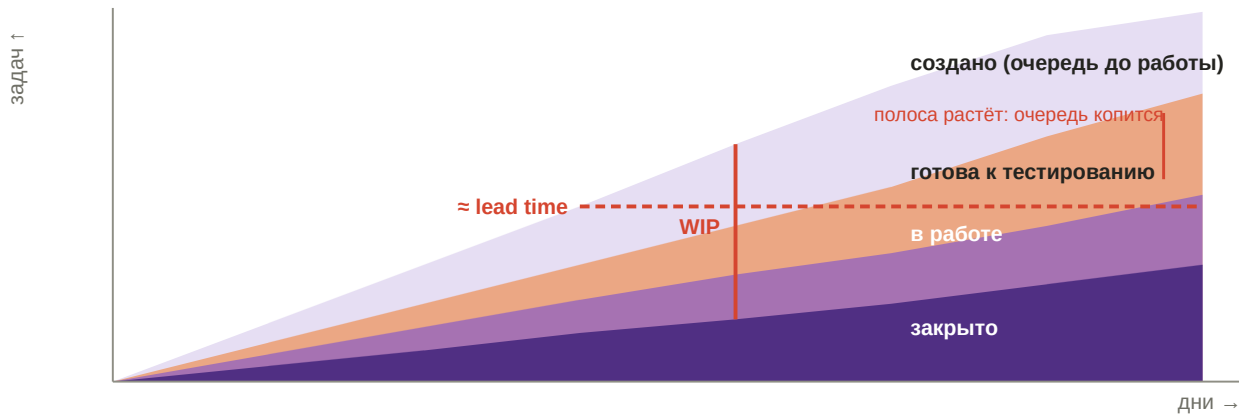


Схема: полосы — статусы, вертикальный отрезок — WIP на этот день, горизонтальный — сколько дней задача идёт от создания до закрытия (приближённый lead time). Расширяющаяся оранжевая полоса — очередь перед тестированием растёт.

Что видно

Вертикаль между верхней и нижней линией — сколько задач сейчас открыто (WIP). **Горизонталь** на одном уровне — сколько дней прошло от создания до закрытия N-й задачи: приближённый lead time. **Толщина полосы** — сколько задач в этом статусе.

Как читать

Полосы параллельны — поток стабилен: сколько входит, столько выходит. Полоса **расширяется** — в этот статус входит больше, чем выходит: здесь копится очередь. Полоса **схлопывается** — очередь разобрали. Верхняя линия растёт, нижняя плоская — набираем работу, не закрывая.

Что делать в Vanilitik

Сузьте окно дат до подозрительных недель: на 4 месяцах мелкие расширения не видны. Поставьте линейку на двух датах: приложение покажет Δ задач и Δ дней между ними. Переключите створ cycle, чтобы проверить, до или после разработки стоит очередь.

Ловушки

CFD по всем командам смешивает потоки: смотрите одну команду или один тип задач. Статус-стоянка («Остановлено») выглядит как очередь — вырежьте её через «минус стоянки». Хвост справа всегда тонкий: последние задачи ещё не успели дойти до конца.

Изломы трендов: как они считаются

Излом — неделя, после которой метрика **устойчиво** сменила уровень, а не просто дёрнулась. Приложение ищет их автоматически в недельных рядах по каждой команде: пропускная способность, lead p50, WIP, эффективность потока. Алгоритм — бинарная сегментация со штрафом, он намеренно простой, чтобы его можно было проверить руками.

1. **Ряд.** Метрика считается по неделям: например, lead p50 задач, закрытых на каждой неделе. Недели без значения пропускаются.
2. **Шум σ .** Берутся разности соседних недель, из их модулей — медиана; $\sigma = \text{медиана} \div 0,6745 \div \sqrt{2}$. Так один выброс не раздувает «нормальный» разброс: это робастная оценка, а не обычное стандартное отклонение.
3. **Лучший разрез.** Для каждой возможной точки k считается, насколько уменьшится сумма квадратов отклонений, если ряд разбить на «до k » и «после k », каждый со своим средним. Берётся k с наибольшим выигрышем; сегменты короче трёх недель не рассматриваются.
4. **Штраф.** Разрез принимается, только если выигрыш больше $2\sigma^2 \ln n$ (штраф в духе BIC): чем короче ряд и чем шумнее данные, тем сильнее должен быть сдвиг. Иначе ряд считается одним куском.
5. **Рекурсия.** Та же процедура повторяется слева и справа от принятого разреза, не больше пяти изломов на ряд.
6. **Фильтр.** Для каждого излома сравниваются средние соседних сегментов; показываются только сдвиги не меньше $1,5\sigma$. В таблице это колонка «Сдвиг»: 5σ — почти наверняка смена режима, $1,6\sigma$ — на грани, смотрите глазами.
7. **Рядом в журнале.** К излому подбирается ближайшее событие команды в окне от 16 дней до него до 5 дней после (событие обычно предшествует эффекту). Нет события — «в журнале объяснения нет»: самая полезная строка таблицы, потому что показывает, чего команда о себе не записала.

Что такое σ . Это «обычный шум» ряда: насколько метрика дёргается от недели к неделе сама по себе, без всяких событий. Считается не как школьное стандартное отклонение (его раздул бы один выброс), а робастно: берутся модули разностей соседних недель, из них медиана, и она делится на 0,6745 и на $\sqrt{2}$. Делители переводят медиану разностей в привычную σ нормального распределения: 0,6745 — потому что медиана модуля нормальной величины равна $0,6745\sigma$, $\sqrt{2}$ — потому что разность двух соседних недель вдвое шумнее одной. Дальше σ — линейка: сдвиг в колонке «Сдвиг» измеряется в этих единицах. 1σ — «в пределах обычного дёрганья», $1,5\sigma$ — порог, с которого излом вообще показывают, 3σ и выше — уровень сменился почти наверняка, 5σ — сомнений нет. Та же σ входит в штраф за разрез: чем шумнее ряд, тем сильнее должен быть сдвиг, чтобы его сочли изломом.

Пример. Lead p50 по неделям: 17, 16, 18, 17, 17, 23, 22, 24, 23. Разности по модулю: 1, 2, 1, 0, 6, 1, 2, 1; медиана 1 $\rightarrow \sigma \approx 1,05$. Лучший разрез между пятой и шестой неделей: средние 17,0 и 23,0, сдвиг $6 \approx 5,7\sigma$, выигрыш далеко больше штрафа. Излом принят; в журнале на той неделе «тестировщик в отпуске» — гипотеза, которую надо проверить по полосе «Готова к тестированию» на CFD.

Почему не «просто сравнить две недели»: одна неделя может быть пустой или с двумя задачами. Сегментация сравнивает средние целых отрезков и требует, чтобы новый уровень держался минимум три недели, поэтому праздники и один тяжёлый эпик изломом не считаются.

Как читать цифры и не обмануться

Перцентили вместо среднего

Среднее время задачи — 12 дней — может значить «все по 12» или «половина по 3, а четверть по 40». Медиана (p50) говорит, как идёт типичная задача; p85 — какое обещание можно дать заказчику; разрыв между ними — размер хвоста. Решение принимается по p85, диагноз ставится по хвосту.

Очередь, а не работа

В большинстве команд задача делается 30–40% своего времени, остальное ждёт: аналитика, ревью, тестирование, окно выкладки. Поэтому «ускорить разработчиков» почти никогда не сокращает lead time, а «не копить готовое перед внедрением» — сокращает. Смотрите на доли работы и очередей раньше, чем на скорость людей.

Размер задачи

Крупные задачи дольше делаются, дольше тестируются, чаще возвращаются и дольше ждут выкладки. Рост lead по неделям без изменений в команде почти всегда означает рост размера задач. Дробление до 2–3 дней разработки — самый сильный рычаг после WIP-лимита.

Событие рядом с изломом — не причина

Двойник показывает, что метрика изменилась на той же неделе, что произошло событие. Это гипотеза, которую надо проверить другими экранами: если отпуск тестировщика действительно причина, вырастет очередь именно на тестирование, а не всё подряд. Один экран — подсказка, три экрана — находка.

Команды нельзя сравнивать в лоб

Разные типы задач, разные компоненты, разная доля ошибок. Сравнивать можно команду с собой месяц назад, или одинаковые типы задач, или долю очередей — относительную величину. «У Альфы lead 17, у Беты 23» ничего не значит без ответа, из чего состоят эти дни.

Чего не делать. Не мерить людей часами и закрытыми задачами (получите списания «для галочки» и дробление ради счётчика). Не считать story points за неделю производительностью (оценка растёт вместе со спросом на неё). Не делать выводов по трём точкам. Не чинить цифру — чинить причину, которую она показала.

Что мерить на каждом уровне

Кому	Главные вопросы	Метрики
Команда на Дейли	Что застряло прямо сейчас и кто может помочь? Что вернулось с проверки и почему? У кого открытых задач больше, чем можно вести?	Старение WIP, WIP на человека, возвраты за неделю
Скрам-мастер на ретро и между спринтами	Где системное препятствие, а не разовая проблема? Что изменилось после договорённости на прошлом ретро? Какой один эксперимент взять на спринт?	Время в статусе, эффективность потока, CFD, импакт событий «до → после», изломы
Владелец продукта на Обзоре спринта и планировании	Сколько реально доходит до заказчика за спринт? Какое обещание по срокам честно дать? Когда закроется эпик и что уходит в прод без проверки?	Пропускная способность, lead p85, эпики по времени и прогноз P50/P85, «минуя этапы»
Тимлид / РМ еженедельно	Куда движется lead и пропускная? Что произошло на неделе и что это изменило? Какое обещание можно дать?	Lead p50/p85 по неделям, CFD, журнал и импакт, изломы, эффективность потока
Руководитель по эпикам и кварталам	Когда закроются эпики и с какой вероятностью? Какой рычаг даст больше дней? Где организация теряет время системно?	Эпики по времени, Монте-Карло P50/P85/P95, what-if, доля очередей по всем командам

Еженедельный ритм: 15 минут

- **Обновить данные.** Дозагрузить выгрузку за неделю (дубли отсекаются сами), «🔄 Пересчитать».
- **Пять экранов.** KPI, работа vs очереди, CFD за последние 4 недели, старение WIP, возвраты. Записать одну фразу: где стоит очередь. Старение WIP удобно открывать прямо на Дейли.
- **Журнал.** Дописать 1–3 события недели: кто вышел, что сломалось, что решили на ретро, что показали на Обзоре спринта. Без журнала через месяц изломы будет нечем объяснять.
- **Один эксперимент.** Один рычаг (WIP-лимит, дробление, окно выкладки) и экран, которым его измеряют через две недели тем же методом.
- **Откуда знаем.** Любую цифру, которая идёт в решение, раскрутить до событий: «*f*» у плитки или запрос к базе.

Исходные данные: три файла из трекера

Двойник строится из трёх CSV-выгрузок. Ниже первые пять строк каждого файла из данных воркшопа (команда Дельта), чтобы было видно, что именно приложение превращает в события. Разделитель `;`, кодировка UTF-8, даты `ДД.ММ.ГГГГ ЧЧ:ММ`; порядок колонок не важен, вид файла распознаётся по заголовкам.

flow.csv

переходы статусов — одна строка на переход. Обязательны «Код», «Новый статус», «Время события»; «Предыдущий статус» и «Время входа» дают точное время в статусе, «Тип» и «Родитель» — иерархию (эпик → задача → подзадача), «Команда» и «Исполнитель» — срезы.

колонки 1–7

Команда	Спринт	Тип	Эпик	Код	Тема	Релиз
Дельта	Спринт 01	User Story	ЕPIC-Платежи	DEL-1001	[Платежи] Массовое редактирование	R-26.6
Дельта	Спринт 01	User Story	ЕPIC-Платежи	DEL-1001	[Платежи] Массовое редактирование	R-26.6
Дельта	Спринт 01	User Story	ЕPIC-Платежи	DEL-1001	[Платежи] Массовое редактирование	R-26.6
Дельта	Спринт 01	User Story	ЕPIC-Платежи	DEL-1001	[Платежи] Массовое редактирование	R-26.6
Дельта	Спринт 01	User Story	ЕPIC-Платежи	DEL-1001	[Платежи] Массовое редактирование	R-26.6

колонки 8–14 (с повторением «Код»)

Код	Предыдущий статус	Время входа в статус	Новый статус	Время события	Длительность	Исполнитель	Родитель
DEL-1001	Открыта	28.05.2026 09:00	Формирование ТЗ	29.05.2026 11:50	27	Лебедев И.	ЕPIC-Платежи
DEL-1001	Формирование ТЗ	29.05.2026 11:50	ТЗ готово	01.06.2026 13:34	74	Лебедев И.	ЕPIC-Платежи
DEL-1001	ТЗ готово	01.06.2026 13:34	Оценка и декомпозиция	02.06.2026 10:28	21	Лебедев И.	ЕPIC-Платежи
DEL-1001	Оценка и декомпозиция	02.06.2026 10:28	Готова к разработке	03.06.2026 10:01	24	Лебедев И.	ЕPIC-Платежи
DEL-1001	Готова к разработке	03.06.2026 10:01	Разработка	04.06.2026 12:43	27	Лебедев И.	ЕPIC-Платежи

team_events.csv

журнал событий команды: дата, категория (процесс, состав, инцидент, внешнее), текст. Объясняет изломы метрик.

Команда	Дата	Категория	Событие
Дельта	24.07.2026	состав	Новый разработчик на компоненте «Платежи», первые недели — онбординг
Дельта	27.08.2026	процесс	Ретро: ввели обязательное код-ревью перед передачей в тестирование
Дельта	14.09.2026	внешнее	Регулятор изменил формат QR, срочная доработка

worklog.csv

списания времени: подзадача (или задача), исполнитель, дата, часы («2,5», «1ч 30м», «2h 15m», «1д» = 8 ч).

Подзадача	Исполнитель	Дата	Часы
DEL-1001-S1	Миронова А.	01.06.2026	5,5
DEL-1001-S2	Лебедев И.	05.06.2026	5,0
DEL-1001-S3	Орехов В.	09.06.2026	4,0
DEL-1001-S3	Орехов В.	10.06.2026	3,5
DEL-1001-S3	Орехов В.	11.06.2026	5,5

Словарик

Lead time	дни от создания задачи до закрытия, включая ожидание; взгляд заказчика.
Cycle time	дни от первого рабочего статуса до закрытия; взгляд команды.
p50 / p85	медиана и 85-й перцентиль: половина задач быстрее p50, 85% быстрее p85.
WIP	work in progress: сколько задач открыто одновременно.
Пропускная способность	сколько задач команда закрывает в неделю.
Эффективность потока	доля времени, когда задачу делали, а не ждали: работа ÷ (работа + очереди).
Очередь (wait)	статус, в котором задача лежит и ждёт следующего этапа.
Стоянка	статус-пауза внутри работы («Остановлено», «Ожидание»); её можно вырезать из cycle time.
CFD	накопительная диаграмма потока: число задач в каждом статусе день за днём.
Возврат	переход задачи назад по потоку: переделка.
Событие журнала	запись о жизни команды: отпуск, ретро, инцидент, внешнее ограничение.
Импакт	метрики за окно до события и после него.
Излом	неделя, после которой метрика устойчиво сменила уровень.
σ (сигма)	обычный шум ряда: типичный скачок метрики от недели к неделе без причин; сдвиг излома измеряется в σ , порог показа 1,5 σ .
Pack	порядок статусов, их категории (работа / очередь / терминал) и workflow по типам задач; выводится из данных.
Провенанс	откуда взялась цифра: цепочка событий, из которых она посчитана.
Двойник	граф событий команды, над которым считаются все проекции; ничего не удаляется и не правится задним числом.

Проверка любой находки. Назовите её статусом и числом («задачи ждут в “Готова к тестированию” по 7 дней с 8-й недели»), покажите тремя экранами, объясните событием журнала и раскрутите до событий. Если хоть одного шага нет, это ещё не находка, а впечатление.