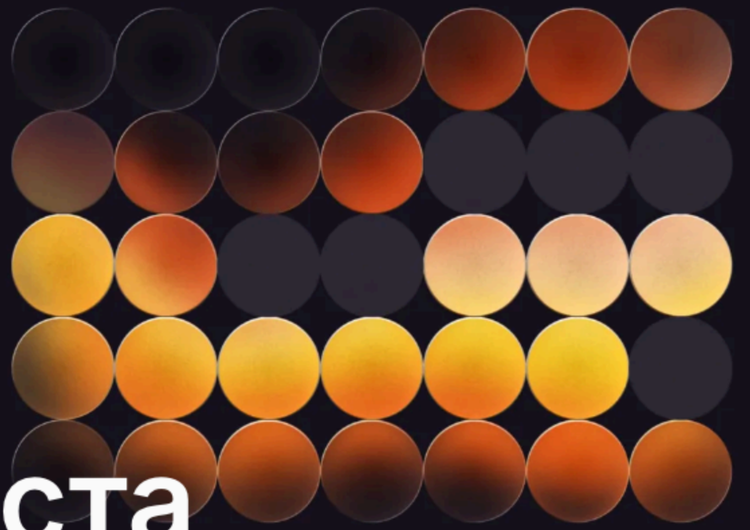
Turbo ML Conf · 18 июля 2026



State of AI4SDLC: AI сдвигает узкие места разработки

Кодинг ускоряется первым. Delivery — только после перестройки всего потока.

TURBO ML CONF

Узкое место переезжает

Главный тезис Flow

Локальное ускорение кода создаёт очередь дальше по потоку

Загрузка схемы...



Часть 1

01

Кодинг ускорился первым

AI снимает локальное ограничение и подсвечивает следующий этап потока

AI перестал быть экспериментом в IDE

Теперь нужно управлять новой физикой разработки

Контекст Shift

Массовое внедрение

default tool

Код, тесты, docs, объяснения.

Неравный эффект

task-dependent

Кодинг ускоряется раньше delivery.

Новый вопрос

next constraint

Где окажется следующее ограничение?

Откуда мы знаем, что происходит

Исследования Данные

Индустриальные отчеты и AI4SDLC дают один сигнал

Индустриальные исследования

- DORA и AI-assisted development reports.
- Кейсы: Google, Microsoft, GitHub, Meta, Uber.

Исследование AI4SDLC

- 50+ исследований 2023–2025.
- Опрос инженеров и техлидов в России.

Сигналы Signals

Внедрение высокое, доверие ниже

Сигналы повторяются в исследованиях

Повседневный инструмент

AI стал частью рабочего дня.

Эффект неравномерен

Эффект зависит от задач и зрелости процессов.

Поставка отстаёт

Нужны review, tests и evals.

Кодинг быстрее, delivery — нет

Главный тезис Bottleneck

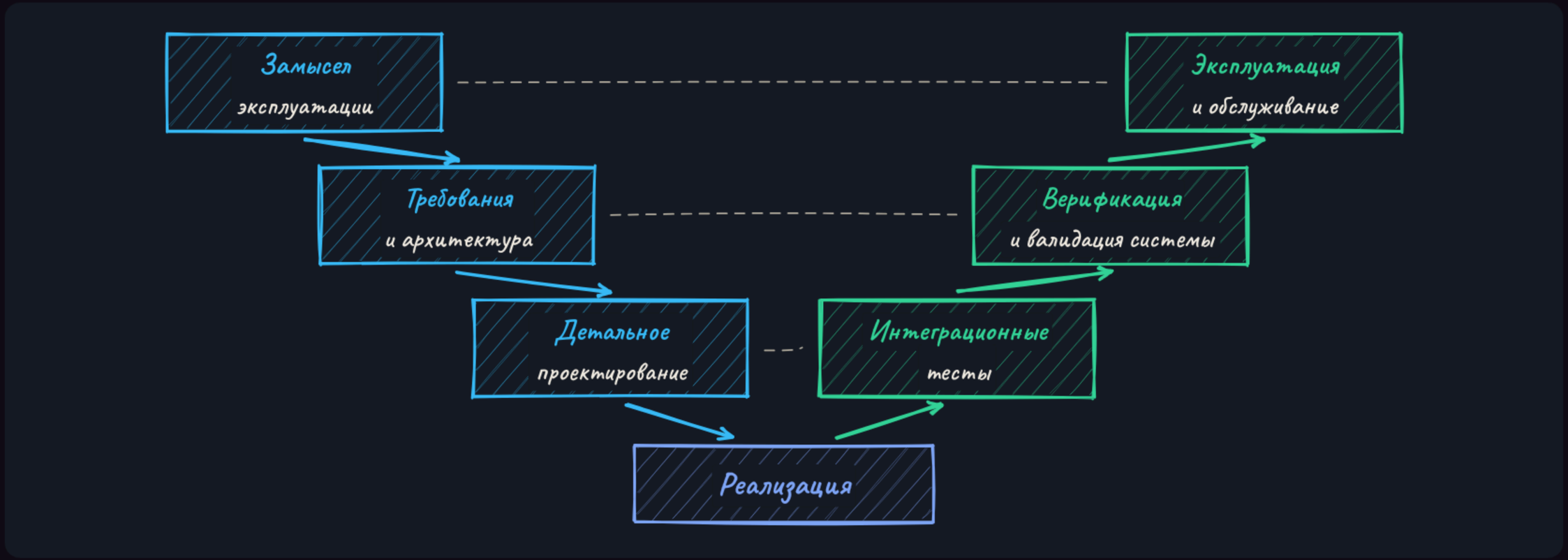
Очередь переезжает в review, тесты и постановку

- Больше WIP перед review
- Тесты поглощают больше изменений
- Слабая постановка масштабирует ошибку
- Проверка становится ограничением

Инженер управляет сквозным процессом

Загрузка схемы...

V-модель связывает требования и проверку





Часть 2

02

Ограничение стало системным

Платформа расширяет throughput без потери контроля

Масштаб задает требования к решению

Контекст Scale

На 10 000+ инженеров нужна платформа

Масштаб

10 000+

Тысячи сервисов, репозиториев и стеков.

Безопасность

fintech

Данные, секреты и PII требуют свой контур контроля.

Экономика

unit economics

Token/infra cost считается на уровне компании.

Метрики North Star

North Star программы: баланс трех осей

Базовая линия, защитные ограничения и владелец у каждой оси

Скорость

speed

Lead/cycle time: от задачи до prod.

Качество

quality

Defects, incidents, rollback, security findings.

Экономика

economics

Tokens, infra и человеко-время vs эффект.

IDP жив:
GUI уходит

Agent-first IDP `GUI → agents`

Контроль остается в платформе

В агент

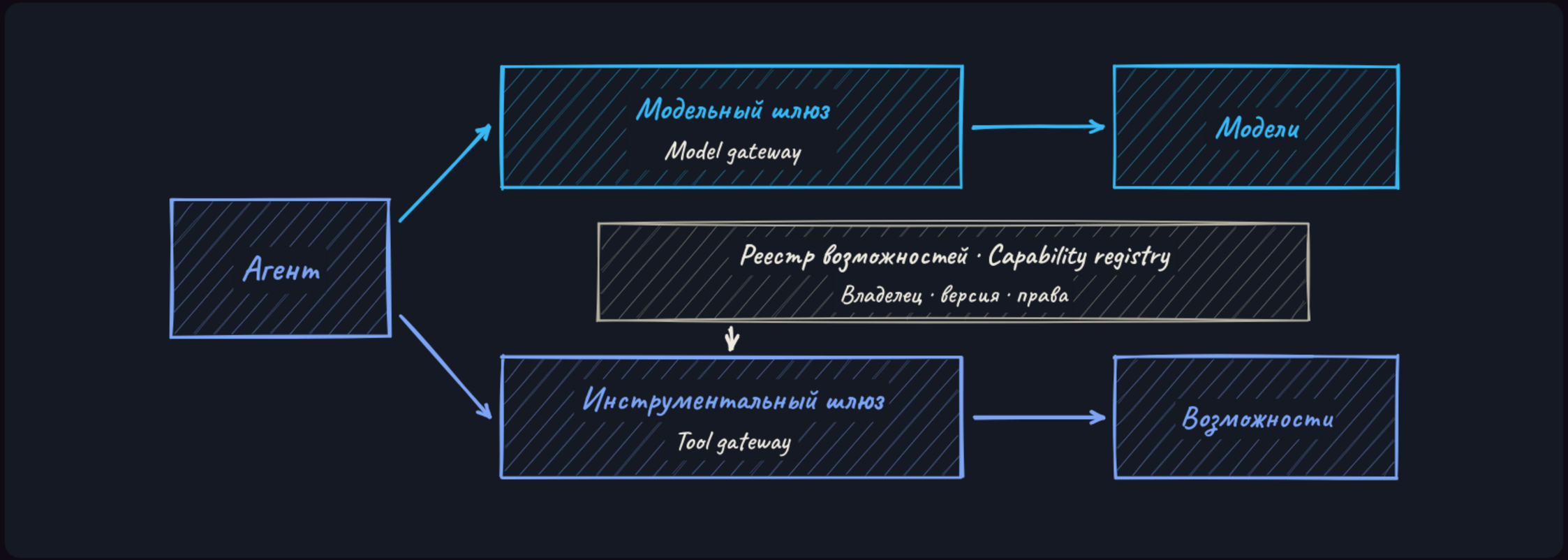
- Intent и следующее действие.
- IDE, wiki, CI/CD, observability.
- Natural language к capabilities.

В платформе

- Доменная логика, policy, state changes.
- Лимиты, квоты, cost, blast radius.
- GUI: настройки, расследования, approvals.

Статья: ↗ tellmeabout.tech

Шлюзы и реестр решают разные задачи



От API к capability map

Platform product Capability map

Capabilities строятся вокруг workflows, не API

Intent map

why

Deploy, observe, secure, govern, approve, remediate.

Machine contracts

how

Tool schema, errors, side effects, cost, policy.

Registry

who owns

Owner, version, scopes, SLA/SLO, evals.

Доверие:
read -> recommend -> act

Trust model `Read → act`

Автономия дается на capability

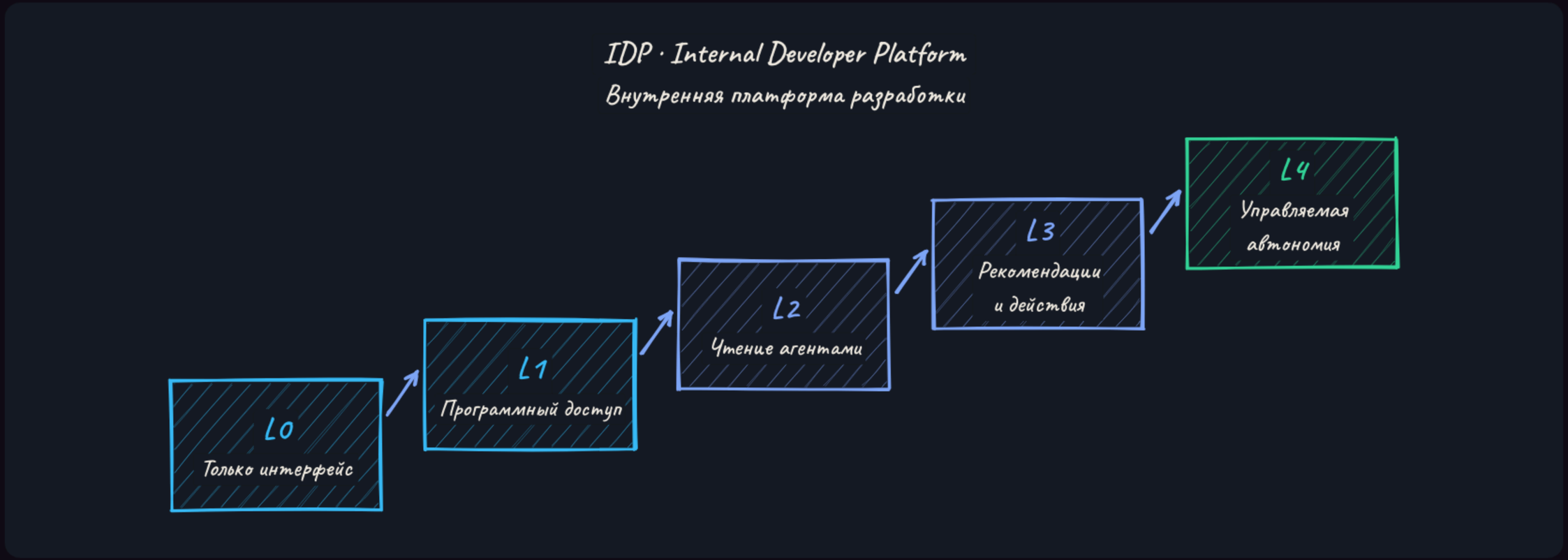
Автономия

- Read: identity, quotas, telemetry.
- Recommend: plan, diff, risk, approval.

Где нужен контроль

- State changes, money or risk.
- Domain semantics: SLO, IAM, data.

Автономия растёт вместе с контролем



Для каждой угрозы нужны защитные меры





Часть 3

03

Агенты снимают очереди

Доменные агенты забирают очереди по всему SDLC

Платформа Модельный шлюз

Model gateway: доступ к LLM

Security и экономика

- One gateway без прямых provider-интеграций.
- Security: secrets, PII, data policy, audit.
- Quotas and cost accounting by team/scenario.
- Models onboard centrally; clients unchanged.

Tool gateway: MCP Hub/CLI

Платформа MCP

Доступ к tools

- CLI, APIs, runbooks, service ops.
- MCP Hub: owner, scopes, schema.
- Controls: allowlist, on-behalf-of, audit, gates.
- Один интерфейс для IDE/agents.

Платформа Nessy

Nessy: company-context copilot

Не просто Copilot

- RAG по codebase, docs, libraries.
- Знает conventions, ADR и golden paths.
- IDE и internal tools.
- Ответы знают текущее состояние систем.

Платформа Spirit

Spirit: agent mode на платформе

Internal PaaS + agents

- VCS/CI/CD: shared policies, paths.
- Agent mode: ticket -> PR.
- Sage: AI-assisted troubleshooting.
- Agents share policy perimeter.

Доменные агенты используют общую платформу





Часть 4

04

Измеряем весь поток

Wait time и rework важнее доли AI-кода

Анти-метрика AI-LOC

AI-код — плохая метрика

AI-LOC не показывает новую очередь

Соблазн

- Легко собрать и показать
- Растёт вместе с внедрением

Вред

- Мотивирует генерировать строки
- Скрывает качество, риски, стоимость

Flow-метрики находят очередь

Метрики DORA × SPACE × DevEx

Wait time, rework, blocked time, safe-change time

Process map

- DORA: delivery и recovery
- SPACE/DevEx: flow и feedback
- Adoption — input, не proof

Signals

- review и pipeline wait
- rework и blocked time
- safe-change time и дефекты

Evals/telemetry вместо MAU

Metrics Telemetry

Ищем блокировки, а не клики

- Трейс: intent -> tools -> approvals.
- Span: user, agent, tool, model, cost.
- KPI: intent success, safe-change time.
- Evals regress agent-facing interfaces.

Reality check Time split

На что уходит время разработчика

Код — лишь часть потока

- Coding: 30–35%.
- Встречи и синки: 25–30%.
- Wiki, docs, context search: 12%.
- Ревью кода и обсуждения: 10–15%.



Часть 5

05

Инженер держит контур

Меньше набора строк, больше intent, orchestration и validation

Социотехника Ключевое наблюдение

Одна платформа — разный эффект

AI внедрение — социотехническая задача

- Одинаковые tools дают разный результат.
- Эффект там, где меняется работа.
- Требования и взаимодействие перестроены.
- Платформа необходима, но недостаточна.

Что отличало успешные команды

Отличие — в перестройке работы

Без эффекта

- Включили ассистента, ждали ускорения.
- Задачи ставятся «по чату».
- Review/tests без изменений.

С эффектом

- Requirements стали agent contracts.
- Review адаптирован к AI-изменениям.
- Context/checks в pipeline.

Люди Сопротивление

Сопротивление — часть внедрения

Automation меняет рабочие процессы

- Fear of replacement реален.
- Работают обучение, прозрачные цели, quick wins.
- Team champions убедительнее приказов сверху.
- Пилот -> обратная связь -> масштабирование.

День инженера: до и после

Практика До / после

Меньше строк, больше intent/context/checks

До

- Пишет код руками.
- Контекст из чатов/wiki/calls.
- Review чужих diffs.

После

- Формулирует intent/context/criteria.
- Оркестрирует agents и parallel tasks.
- Валидирует результат, больше design.

Режим агента зависит от задачи



Самостоятельность требует ясной задачи и проверки

Роль инженера становится шире

Роль Leverage

Ценность — управление системой изменений

Постановка

Цель, context, ограничения, DoD.

Оркестрация

Разложить, delegate, собрать результат.

Валидация

Проверить behavior, risks и intent.

Навыки Skills

Навыки-рычаги для инженера

Чтобы агенты усиливали

- Context engineering: управлять контекстом агента.
- SDD: проверяемые contracts вместо размытых задач.
- Evals: измеримое качество agent work.
- Platform literacy: capabilities and boundaries.

Чего не происходит

Мифы | Reality check

Популярные страхи vs практика

- Инженеры не исчезают: judgment важнее.
- Джуны нужны; меняется обучение.
- Ревью кода проверяет contracts and proof.
- Ответственность остается на человеке.

Пять сдвигов AI4SDLC

Ускоряйте поток, а не отдельный этап

- Кодинг перестал быть главным ограничением
- Очереди переехали в постановку и проверку
- Платформа расширяет безопасную автономию
- Evals и flow-метрики показывают эффект
- Инженер оркестрирует и валидирует изменения

- Ускоряйте поток, а не отдельный этап

AI4SDLC

Исследование и материалы

Исследование, материалы и продолжение дискуссии — в канале AI4SDLC.

Александр Поломодов

Technical Director & Fellow,
T-Технологии



@AI4SDLC