

— KEYNOTE · ОПЫТ

УПРАВЛЕНИЕ КОНТЕКСТОМ

при растущей кодовой базе автотестов /
с помощью ИИ

// SPEAKER · 2026



— КТО Я?

Никита Руленко



Ведущий инженер производительности

CONTEXT RECOVERY

6+ ЛЕТ В AI·QA

— ВОПРОС · В ЗАЛ

Для кого этот доклад?

— ЧЕТЫРЕ ТИПА · В ЗАЛЕ

01 **QA**

02 **Dev**

03 **Load**

04 **Manager**

ЖАЛОБА · #1 НА ОНБОРДИНГЕ

«Большой проект / много кода»

«Мой агент
делает не то!»

— ДИАЛОГ · 1 ИЗ 2

↓ ОТВЕТ

«Ты не провёл
онбординг!»

«Мой агент
делает не то!»

— ДИАЛОГ · 2 ИЗ 2

↓ ОТВЕТ

«Ты не провёл
онбординг!»

ТЕЗИС · ФОРМУЛА ОНБОРДИНГА

ТОЧНЕЕ

CAUSE · ЧТО ВКЛАДЫВАЕМ

ОНБОРДИНГ

ЗНАЧИТ



МЕНЬШЕ

EFFECT · ЧТО ПОЛУЧАЕМ

ЗАТРАТ

В МЕСЯЦ · ПО РЕГИОНАМ

ЕВРОПА

€7-10k

на разработчика

США

\$20k

на разработчика

РОССИЯ

100-
250k ₺

на разработчика

АЗИЯ

\$3-5k

на разработчика

— ДЕМО-ПРОЕКТ · МАСШТАБ

• BEAN & BREW

СОТНИ

ТЕСТОВ

ТЫСЯЧИ

СТРОК КОДА

STACK / FRONT · BACK · DB

МЕТАФОРА • КАК НАРЕЗАТЬ МОРКОВКУ

Подходы к памяти бывают разной сложности

ПРОСТОЙ

СЛОЖНЫЙ

01 • НОЖ



MD files

PLAIN TEXT

02 • ТЁРКА



GitHub Issues

СТРУКТУРА

03 • КОМБАЙН



Mem0

VECTOR SEARCH

04 • ЛАЗЕР



Helix MCP

KNOWLEDGE GRAPH

МЕТАФОРА • КАК НАРЕЗАТЬ МОРКОВКУ

Подходы к памяти бывают разной сложности

ПРОСТОЙ

СЛОЖНЫЙ

01 • НОЖ



MD files

PLAIN TEXT

02 • ТЁРКА



GitHub Issues

СТРУКТУРА

03 • КОМБАЙН



Memo

VECTOR SEARCH

04 • ЛАЗЕР



Helix MCP

KNOWLEDGE GRAPH

МЕТАФОРА • КАК НАРЕЗАТЬ МОРКОВКУ

Подходы к памяти бывают разной сложности

ПРОСТОЙ

СЛОЖНЫЙ

01 • НОЖ



MD files

PLAIN TEXT

02 • ТЁРКА



GitHub Issues

СТРУКТУРА

03 • КОМБАЙН



Memo

VECTOR SEARCH

04 • ЛАЗЕР



Helix MCP

KNOWLEDGE GRAPH

МЕТАФОРА • КАК НАРЕЗАТЬ МОРКОВКУ

Подходы к памяти бывают разной сложности

ПРОСТОЙ

СЛОЖНЫЙ

01 • НОЖ



MD files

PLAIN TEXT

02 • ТЁРКА



GitHub Issues

СТРУКТУРА

03 • КОМБАЙН



Mem0

VECTOR SEARCH

04 • ЛАЗЕР



Helix MCP

KNOWLEDGE GRAPH

— ОСНОВА · АГЕНТ ЧИТАЕТ ПЕРВЫМ

// NOTE
BASE · ROOT

МВД — **ЭТО**
БАЗА



MD-файлы дают **правила**

✓ Хорошо

- Разделение файлов по ролям
- Документация известных проблем
- Бесплатно

✗ Плохо

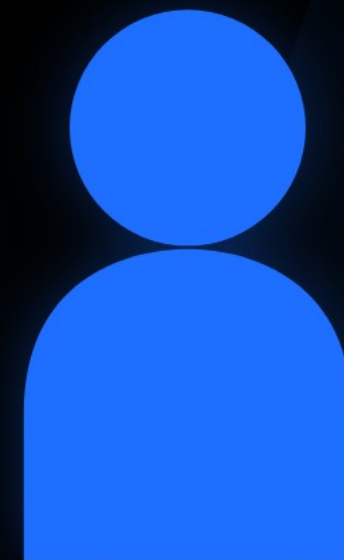
- Всё в одном файле и он большой
- Слишком много файлов



Читает файлы



• READ ONLY · PROMPT CONTEXT



AGENT

Рецепт хорошего промта

10 XML-блоков, которые превращают набор инструкций в воспроизводимый онбординг.

PROMPT.XML

BLOCK 01 · ROLE

```
<role>
```

```
    КТО АГЕНТ И В КАКОМ ТОНЕ ОТВЕЧАЕТ
```

```
</role>
```

→ задаёт персону и тон живого инженера

PROMPT.XML

BLOCK 02 · TASK

<task>

ЧТО СДЕЛАТЬ И РАДИ КАКОЙ ЦЕЛИ

</task>

→ фиксирует однозначную цель сессии

PROMPT.XML

BLOCK 03 · CAPABILITIES

```
<approach_capabilities>  
  набор подходов и практик  
</approach_capabilities>
```

→ какие техники и практики агент должен использовать

PROMPT.XML

BLOCK 04 · SOURCES

<context_sources>

откуда читать и что вызывать запрещено

</context_sources>

→ граница между разрешёнными и запрещёнными источниками

```
<execution_protocol>
```

фазы работы по порядку — до финиша

```
</execution_protocol>
```

→ линейный план: чтение → свёртка → отчёт

PROMPT.XML

BLOCK 06 · SURFACE

```
<what_to_surface>
```

```
    какие наблюдения — сигнал, а какие —  
    шум
```

```
</what_to_surface>
```

→ отсекает дампы тикетов и таблицы без выводов

PROMPT.XML

BLOCK 07 · WRITES

```
<writes_policy>
```

ЧТО МОЖНО МЕНЯТЬ В РЕПО И КОГДА

```
</writes_policy>
```

→ границы автономии: update, close, add_comment, edit

PROMPT.XML

BLOCK 08 · REPLY

<reply_discipline>

рамки первой и последней строки ответа

</reply_discipline>

→ убирает пролог и эпилог из ответа модели

PROMPT.XML

BLOCK 09 · OUTPUT

```
<output_format>  
  структура отчёта для инженера  
</output_format>
```

→ фиксирует разделы, порядок и метрики

PROMPT.XML

BLOCK 10 · EXAMPLE

<example>

эталон хорошего и антипример

</example>

→ модель видит, к чему стремиться и чего избегать

PROMPT.XML

BLOCK 10 · EXAMPLE · FILLED

<example>**Хорошие bullets для «Что MD дал и где не сошёлся»:**

- «Все **7 MD** сходятся на **336 тестах** – расхождений между файлами нет»
- «MD говорит **handler=67%**, по факту **71%** – MD устарел на **4 п.п.**»
- «Last updated везде **2026-04-14** (3 дня назад) – данные свежие»

Плохие bullets:

- «AGENTS.md → docs/known_issues.md (ребро 1 из 8)» – *таблица без вывода*
- «Score 0.83 для test-context.md» – *это не из MD-подхода*

</example>

→ модель видит, к чему стремиться и чего избегать — на живом примере

ISSUE 42



✓ CONTEXT

— ПОДХОД · ТИКЕТЫ

GitHub Issues

ВТЯГИВАЕМ КОНТЕКСТ → ИЗ ТИКЕТОВ



РАЗБОР · ПЛЮСЫ И МИНУСЫ

MD + GitHub дают **human interface**

✓ Хорошо

- Строгий регламент нейминга
- Ход работы в комментариях
- Бесплатно

✗ Плохо

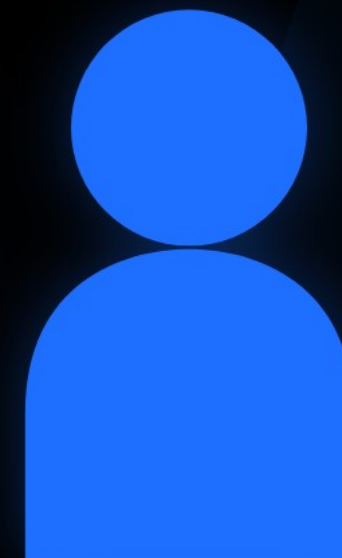
- Одноранговые тикеты
- Отсутствие конвенции нейминга
- Дублирование тикетов



читает через MCP



работает с задачей



AGENT

ФОРМА · AGENT-AGNOSTIC

Тикет — интерфейс контроля

~~TICKETS~~ → VECTOR SEARCH

— ПОДХОД · СЕМАНТИКА

Memo **framework**

семантический поиск → ВМЕСТО кучи тикетов



MD + Memo дают **скорость**

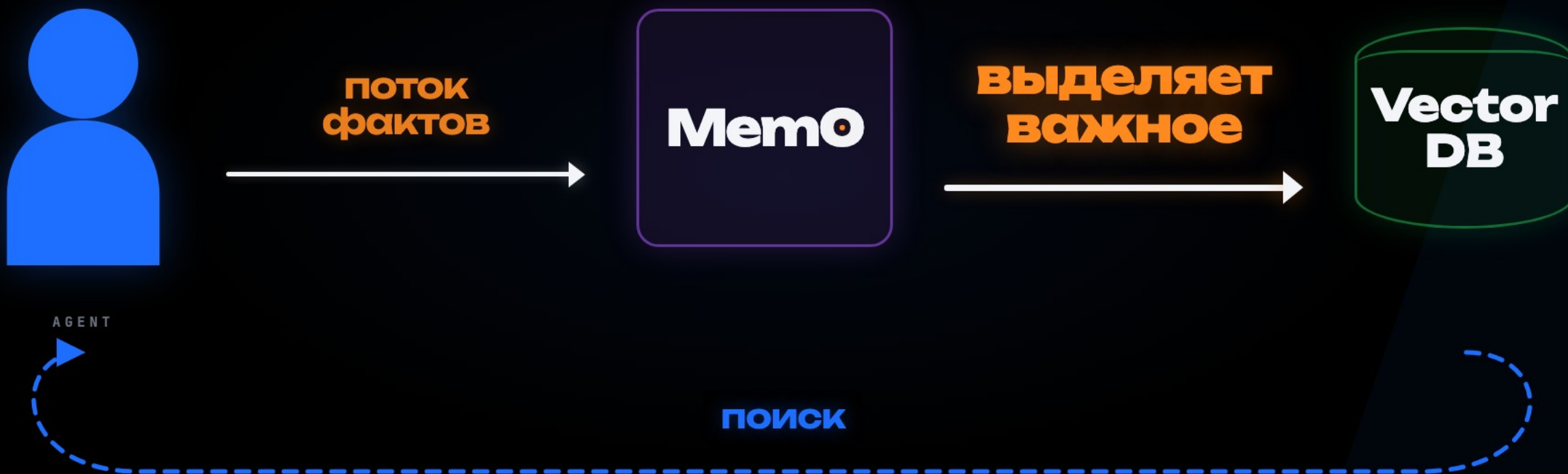
✓ Хорошо

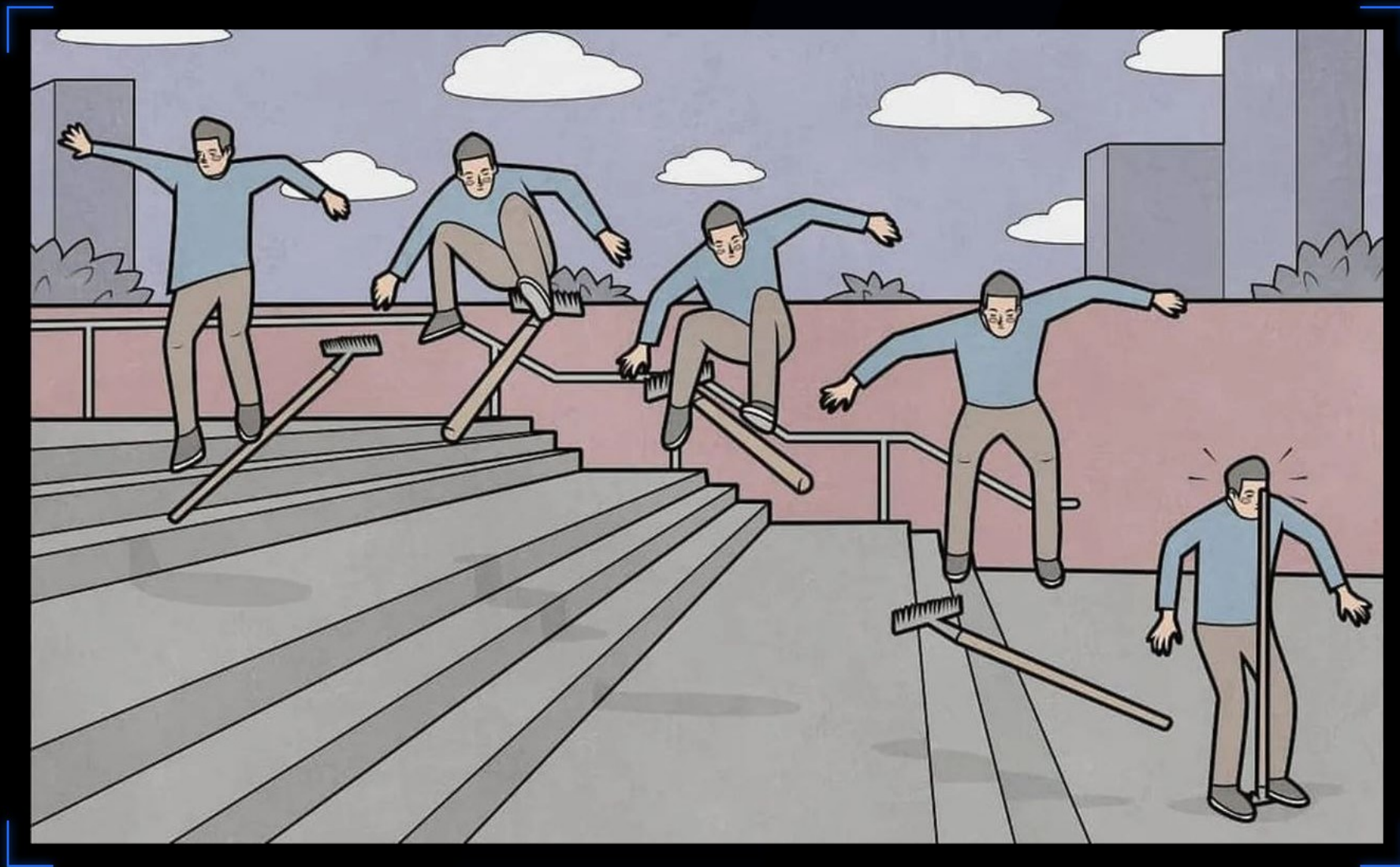
- Семантический поиск
- Самый популярный — большое комьюнити

✗ Плохо

- Может пропускать факты
- Платно — LLM tokens + infra или подписка

MEMO · КАК ЭТО РАБОТАЕТ





• FACT



• CAUSE

— ПОДХОД · ГРАФ

Helixir

MCP

СВЯЗЬ фактов — с причинами



MD + Helixir дают **причинность**

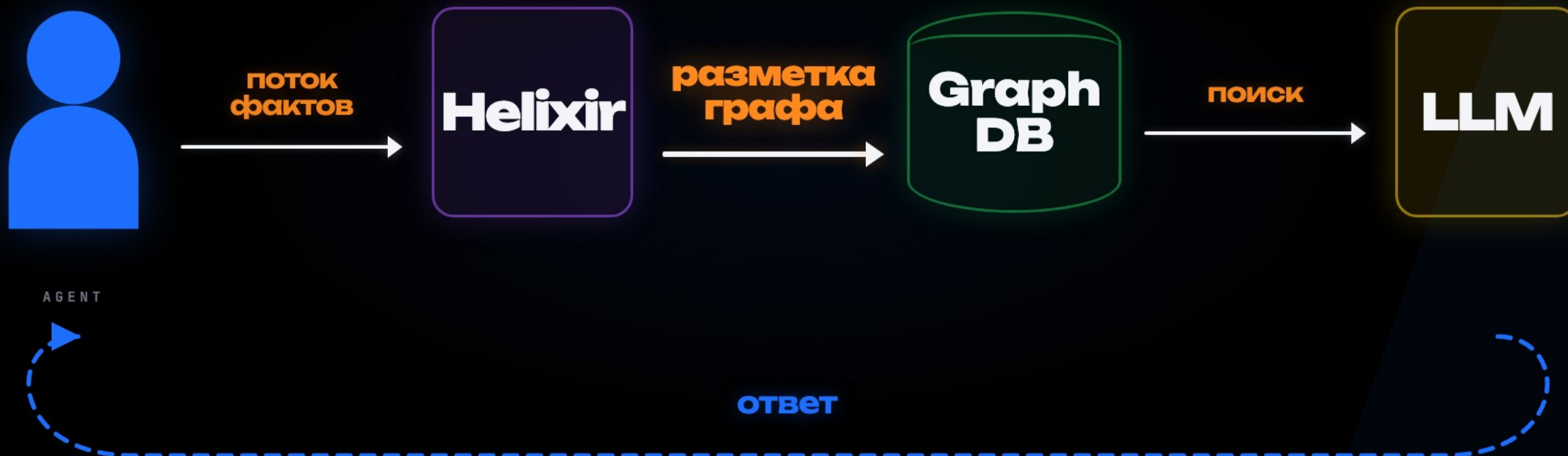
✓ Хорошо

- Причинные цепочки
- Категории фактов
- Размышление **без засорения памяти**

✗ Плохо

- Нет удаления фактов
- Платно и дорого
- Один мейнтейнер – я :)

— HELIXIR · КАК ЭТО РАБОТАЕТ



Как я ЭТО ИЗМЕРИЛ?

— МЕТОД · КАК ИЗМЕРЯЕМ

LLM-as-a-Judge

одна модель → оценивает другую

— ТРИ ГРУППЫ ВОПРОСОВ · ПРОВЕРКИ

01 / SUITE

ФАКТЫ

TEST · ЧТО ЗНАЕТ

02 / SUITE

СВЯЗИ

GRAPH · КАК СВЯЗАНО

03 / SUITE

Понимание решений

RATIONALE · ПОЧЕМУ ТАК

— СРАВНЕНИЕ · 4 ПОДХОДА

Лучшие в своём деле

	ПРОСТОТА	СВЯЗНОСТЬ	AI УДОБСТВО	ПОДДЕРЖКА	ЦЕНА
01 MD files	★	.	.	.	\$0.032
02 GitHub Issues	.	.	★	.	\$0.033
03 MemO	.	.	.	★	\$0.018
04 Helixir	.	★	.	.	\$0.102

★ ЛИДЕР КОЛОНКИ ■ ДЕШЕВЛЕ ■ ДОРОЖЕ

Лидеры инференса

01 → SPEED

Cerebras

GPT-OSS 120B

3000

токенов в секунду

скорость →

02 → PRICE

DeepSeek

DEEPSEEK R1

\$0.55

за 1M входных

цена →

03 → VOLUME

Qwen

QWEN3.6 PLUS

1.4 T

токенов в сутки

объём →

04 → QUALITY

Anthropic

CLAUDE OPUS 4.7

87.6%

SWE-bench Verified

качество →

Разработчики Claude

А ТЕМ ВРЕМЕНЕМ
ГДЕ-ТО В САН-ФРАНЦИСКО



КАДР · СЕРИАЛ «ВО ВСЕ ТЯЖКИЕ» · [imdb.com/title/tt0903747](https://www.imdb.com/title/tt0903747)

Польза

ТЕСТИРОВЩИКАМ

- **ВАНШОТИМ** API тесты
- **ВИДИМ** актуальность тикетов

Польза

разработчикам

- **следим** за кодом
- агент расскажет про легаси

Польза

Менеджеру

- **я вижу** работу команды
- **могу ставить задачи агентам**

— VALUE • БИЗНЕСУ

Польза бизнесу

- сотрудник **быстрее** приступит к работе

Совет

НОВИЧКАМ

- **используй проверенные решения** — Cursor / Claude

— ADVICE • ОПЫТНЫМ

Совет

ОПЫТНЫМ

- **подбери** инструмент под свой проект
- тебе должно быть удобно

NIKITA RULENKO · АВТОР HELIXIR

Спасибо за **ВНИМАНИЕ!**

Все материалы доклада, код бенчмарков и демо-приложение — по QR-кодам ниже.



01 · RESEARCH

Ресерч и бенчмарки

Методология, результаты v2/v3, Context Recovery, сравнение подходов к памяти.

github.com/nikita-rulenko/heisenbug-research



02 · DEMO

Кофейный портал

Go-проект для онбординга: AGENTS.md, cursor rules, онбординг-промты — всё на русском.

github.com/nikita-rulenko/heisenbug-coffee-portal