

Context engineering

Архитектурные подходы

Улучшение результатов применения ИИ через
управление рабочим пространством LLM

Спикер



Екатерина Якуненко

Юрист-эксперт в Нейроюриссте

Автор TG-канала и блога «Делай RAG»

Спикер образовательных программ для юристов

Изучаю тренды внедрения ИИ в российскую юридическую практику

Редактор открытой базы знаний сообщества Нейросети I ilovedocs

Навайбкодила бесплатные проекты для юристов-рекламщиков

Блок первый

Контекст и инженерия

Как растет и умирает контекст,
и причем здесь инженеры

Контекст – это коробка, и большую часть туда положили не вы



системный промпт вендора

ваши постоянные инструкции

описания подключённых инструментов

приложенные документы

результаты поиска в Интернете и вызовов инструментов

история диалога

ваш вопрос

размышления модели

Context rot: деградация контекста

Отравление / Poisoning	галлюцинация/ошибка попадает в контекст и многократно цитируется
Замусоривание / Confusion	лишняя информация используется для генерации низкокачественного ответа + tool loadout
Расфокус / Distraction	контекст разрастается так, что модель over-фокусируется на незначительных деталях
Конфликт / Clash	новая информация/инструменты конфликтуют с уже имеющимися
Lost in the middle	фокус внимания на информации в начале и конце контекстного окна

Drew Breunig, [How long contexts fail](#)



Деградация контекста происходит внезапно и задолго до исчерпания заявленного вендором размера контекстного окна

“

Context engineering refers to the set of strategies for curating and maintaining the optimal set of tokens (information) during LLM inference.

Anthropic, [Effective context engineering for AI agents](#): «набор стратегий по курированию и поддержанию оптимального набора токенов (информации) в ходе инференса»

Блок следующий

Архитектурный (о методах)

- 1 КОНТЕКСТ
НАПОЛЯЕТЕ ВЫ
- 2 НАПОЛНЕНИЕ ЗАРАНЕЕ
ДЕТЕРМИНИРОВАНО
- 3 ЧТО МЕНЯЕТ
АГЕНТНОСТЬ

Несколько простых советов



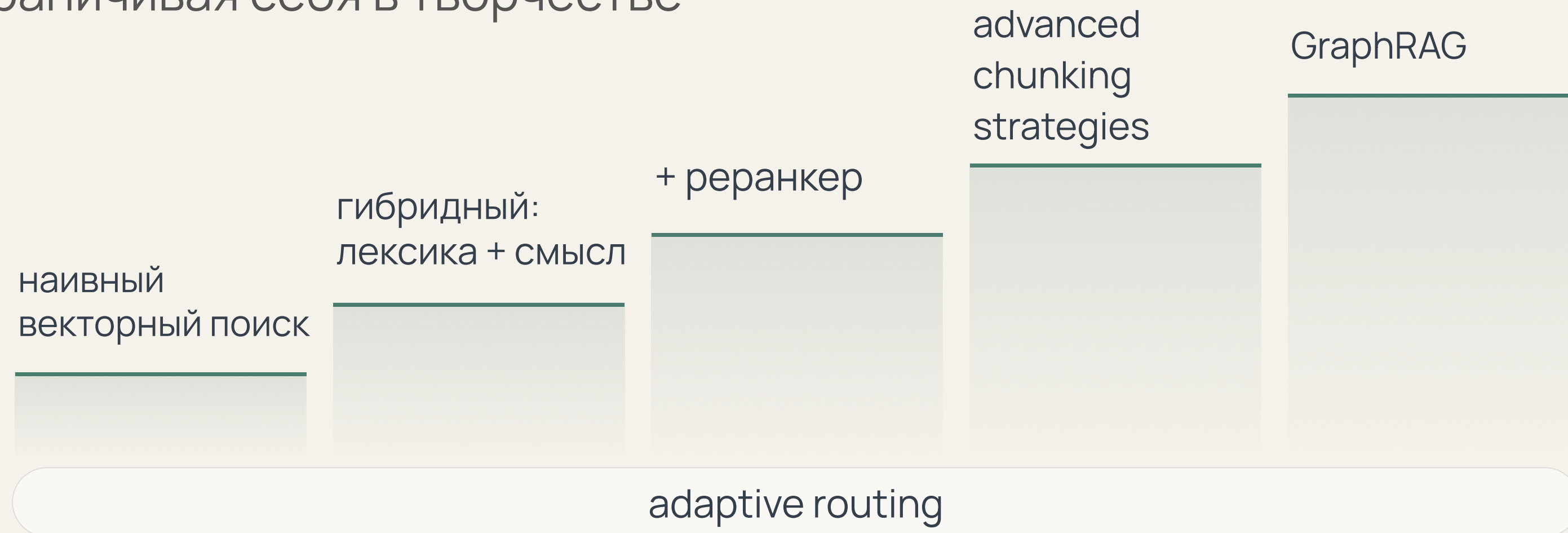
приносим только относимое к задаче
используем функционал проектов
не раздуваем инструкции проектов и описания скиллов
длинные документы вверх, вопрос в конец
завершаем долгий чат просьбой о handoff-файле для нового чата

*Эффективность использования контекста
держится только на вашей дисциплине*

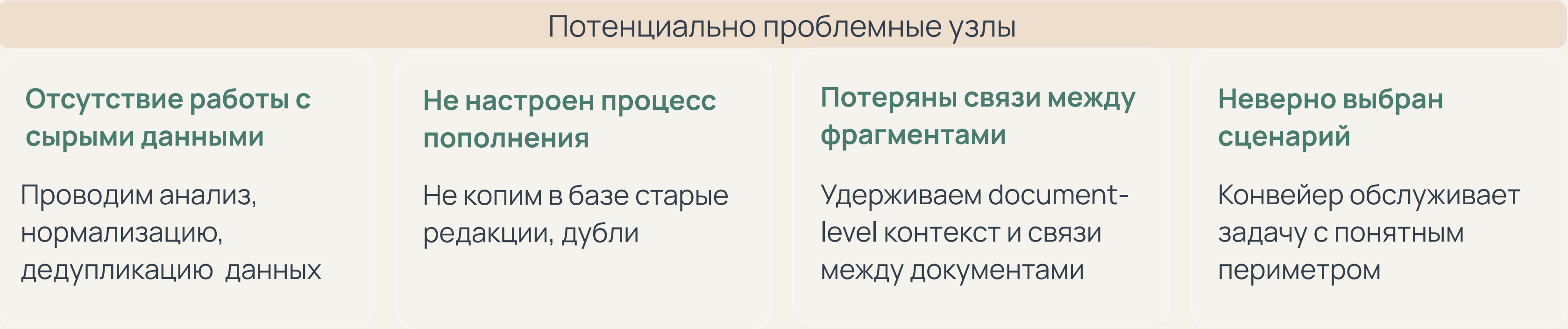
А если команда большая, или данных очень много?

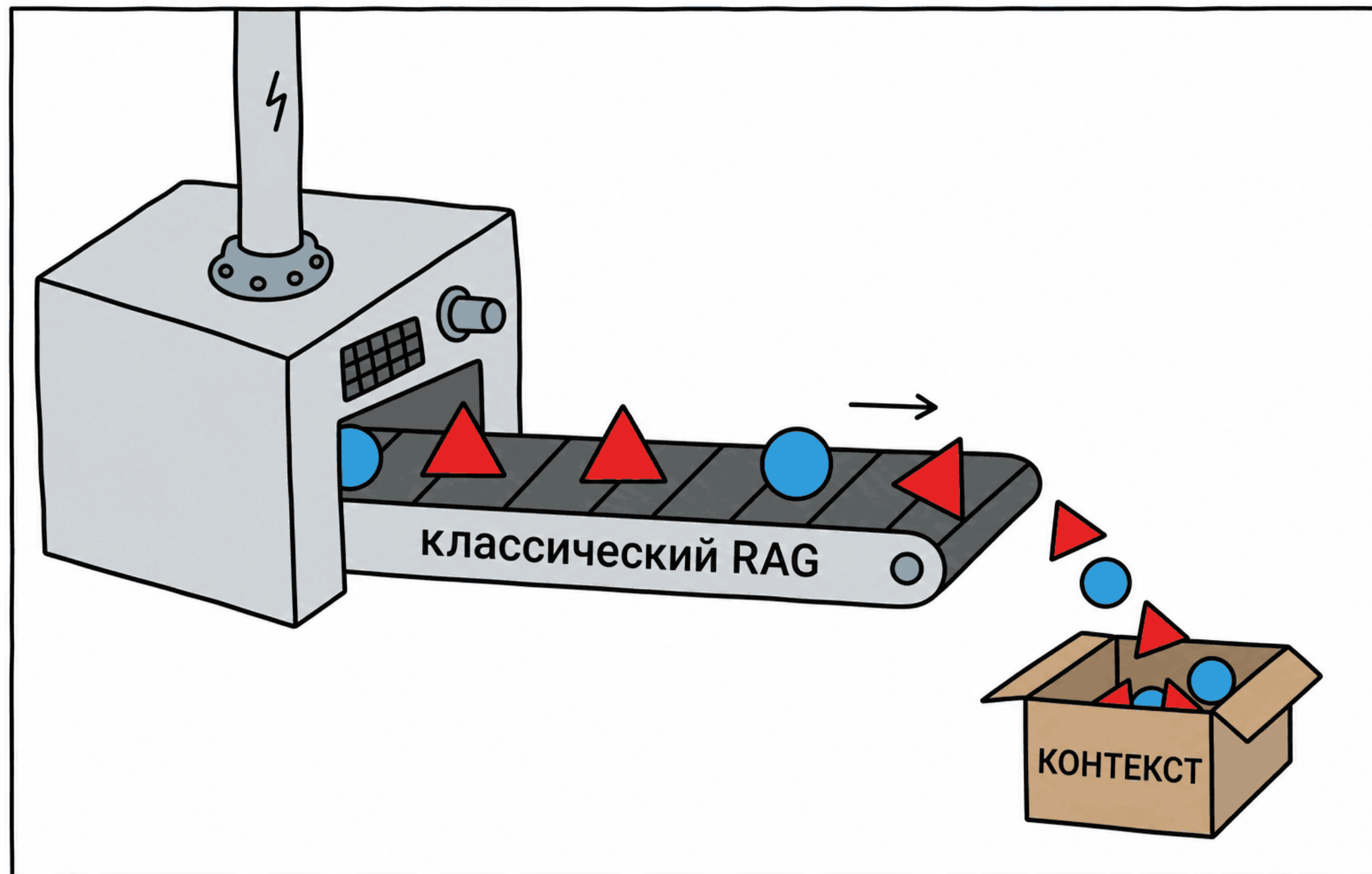
Значит, пора делать RAG!

Не ограничивая себя в творчестве



Где принимаются архитектурные решения





*Даже навороченный RAG быстро достигает потолка при масштабировании:
конвейер проектируется в попытке предугадать вопросы.*

Теперь все делают харнессы



Цикл реализуется в *харнессе*

Агентность подразумевает работу в *цикле*

воспринять → спланировать → действовать →
проверять → повторять

LLM – движущая сила

Инструменты: веб-поиск,
скиллы, коннекторы

Разные харнессы одинаковой модели = разная эффективность

Где принимаются архитектурные решения



01

выбор харнесса
коробочный / свой



02

наполнение инструментами
под конкретный сценарий



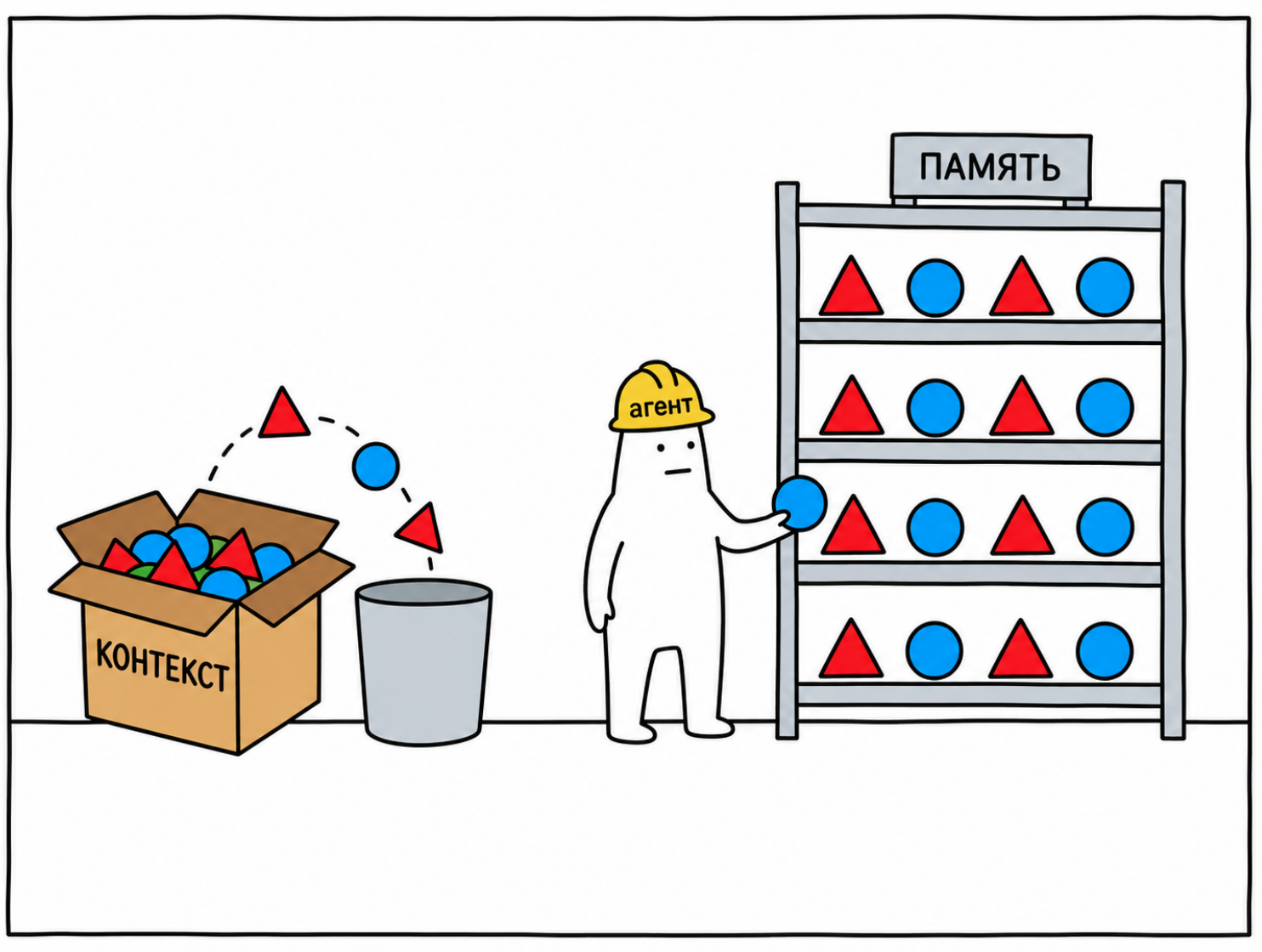
03

проектирование рабочего пространства



04

управление памятью и знаниями



Управление памятью и знаниями

*Память агента переживает ограниченность контекстного окна.
Инженерия контекста становится инженерией знаний: и ваших собственных, и агентных.*

ПОДХОД	КОГДА РАБОТАЕТ
Файлы в папках	один проект, малая команда, прозрачность и версионирование. Предпочтительный формат .md
Векторные базы	большие корпуса данных. RAG теперь – инструмент агента
Графы знаний	связи, время, контроль доступа и аудит

LLM-вики

Знание компилируется один раз и поддерживается актуальным, а не выводится заново на каждый запрос

Почему управление памятью и данными так важно

Просыпаясь, агент ничего не помнит

Каждая сессия начинается с нуля. Что решили в прошлый раз, что уже проверяли — агент этого не знает, даже если оставались артефакты.

AGENT.MD, навигационный слой базы знаний

Сбор контекста агентом не детерминирован

Решая одну и ту же задачу в разных сессиях, агент может по-разному наполнять свой цикл. Кроме того, его суждения об относимости данных непрозрачны.

Структурность и нормализация данных повышает определенность

Риск промпт-инъекций

Входящая документация становится источником дополнительных инструкций. И вы не хотите, чтобы они были вредоносными

Процесс контроля входящей информации

Блок последний

Мотивационный и *стратегический*

Что унести с собой для
обдумывания

Начинать не поздно

1

ПОТРЕБИТЕЛЬ
ЗНАНИЙ

Осознание ценности управления знанием

Knowledge Management
существует давно, но именно
агентный ИИ дал ему заказчика
и измеримый результат.

**Наведение порядка в данных
начинает окупаться.**

2

ЧЕТКИЙ
СЦЕНАРИЙ

Не пытайтесь объять сразу все

Один узкий часто
повторяющийся кусок, где вы
точно знаете правильный ответ:
один процесс, одна папка, один
набор проверочных вопросов.

3

НЕПАХАННОЕ
ПОЛЕ

Методов и рецептов пока нет

Никто пока не знает, как
правильно, особенно для
юридического домена. Россия
ограничена в белом
использовании топовых
инструментов.

У нас есть 5 минут на
вопросы!

*А что не успеем —
читайте в моем канале*

