NEXUS

Инфраструктура доверенного контекста для AI в промышленности

Как безопасно подключать LLM и мультиагентов к производственным данным, знаниям и расчётам

не чат над базами **не AI-first** **не прямой доступ к данным**

AI должен получать не сырые данные, а проверенный, ограниченный и объяснимый производственный контекст.

AI перестал быть экспериментом. Он стал вопросом доступа.

Главный риск 2026 года — не слабая модель, а неуправляемый контур AI-потребления данных.

1. Каким данным верит AI?

Источник, версия, статус подтверждения, качество, lineage, актуальность.

2. Почему AI имеет право их видеть?

Роль, цель запроса, контекст, политики, маскирование, зона доступа.

3. Кто отвечает за вывод?

Модель интерпретирует. Расчёт и факт должны оставаться в детерминированном контуре.

CIO будут помнить не схему, а вопрос: “Можем ли мы доказать, что именно знал AI и почему он имел право это знать?”

Прямой доступ LLM к промышленным данным — архитектурная ошибка

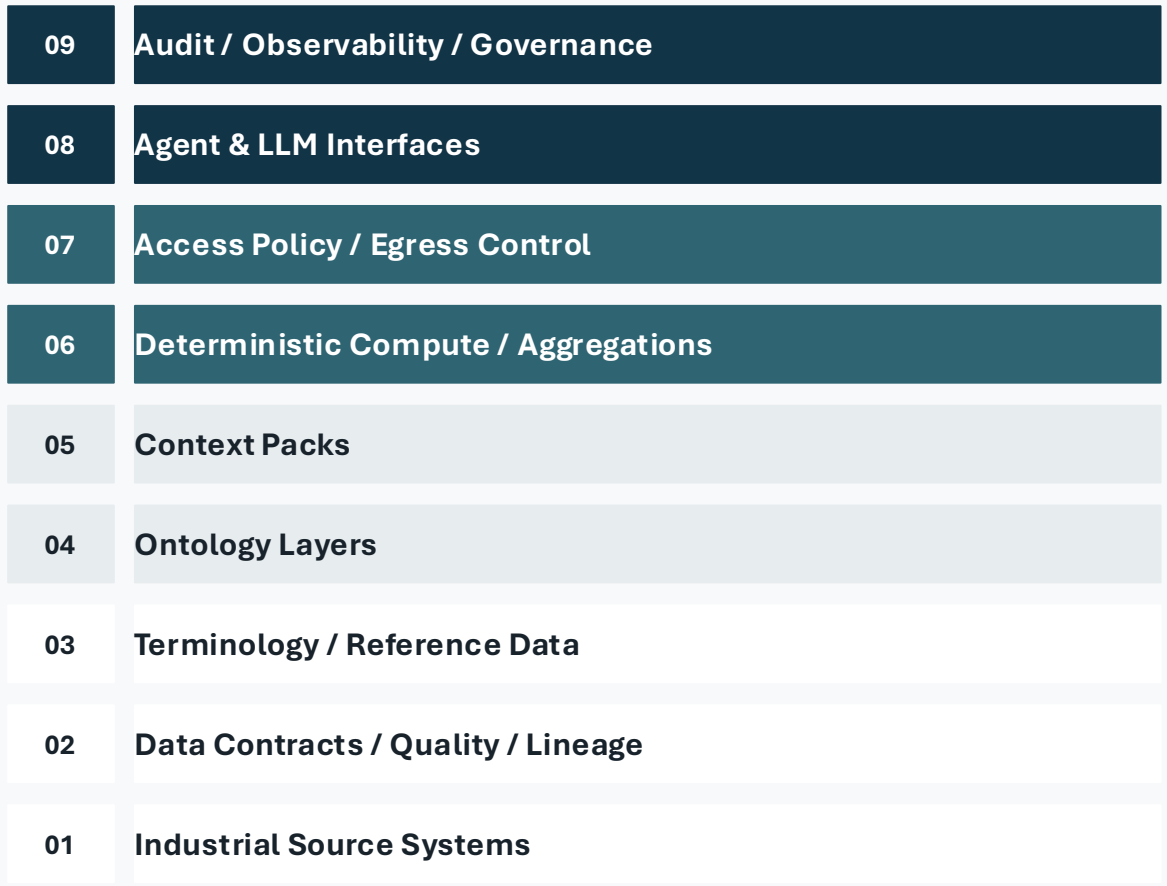
Проблема не в “галлюцинациях” как таковых. Проблема в отсутствии управляемого контекста.

нет статуса факта
нет контролируемых терминов
нет расчётного контура
нет границы доступа
нет воспроизводимого audit trail

Вывод: промышленному AI нужен не “доступ к данным”, а доступ к подготовленному смыслу.

NEXUS — многослойная инфраструктура доверенного контекста

Каждый слой отвечает за свой вопрос: данные, смысл, расчёт, доступ, аудит.



ГЛАВНАЯ ИДЕЯ

LLM не получает “всё”.
LLM получает ровно тот context pack, который разрешён политикой и достаточен для задачи.

Сырые данные остаются в системах-источниках. Истина, расчёт и право доступа не делегируются модели.

Производственные данные должны стать пригодными для AI-потребления

Не каждое поле из MES/PLM/SCADA является фактом, который можно отдавать агенту.

Источник

какая система владеет данными

Время

event time, processing time, timezone

Качество

freshness, gaps, confidence, sensor status

Статус

candidate, confirmed, corrected, locked

Версия

документа, нормы, карты, справочника

Lineage

как получено, кем изменено, где использовано

Сначала доказуемость данных. Потом — AI-интерпретация.

Терминология — это инфраструктура, а не словарь в Excel

Если предприятие не управляет смыслом терминов, LLM будет управлять им за него.

“Справочник” — это договор о смысле.

синонимы и локальные названия: “40Х”, “Сталь 40Х”, “40Х ГОСТ …”
статусы: действующий, заменённый, архивный, запрещённый термин
классификаторы: оборудование, материал, операция, дефект, простой, маршрут
каноническая форма для систем, людей, агентов и отчётов

Без этого слой RAG превращается в поиск похожего текста, а не в работу с промышленным знанием.

Онтологии нужно разделять: отрасль, компания, роль

Один и тот же факт должен иметь разные безопасные представления для разных потребителей.

Общеотраслевая

Базовые понятия отрасли: объект, операция, материал, маршрут, норматив, дефект, риск.

Корпоративная

Классификаторы, регламенты, KPI, оргструктура, версии норм, внутренние правила.

Ролевая / партнёрская

Что разрешено конкретной роли, агенту, клиенту или партнёру в рамках цели запроса.

Слой онтологий отвечает на вопрос не только “что это значит?”, но и “для кого это значит именно так?”.

Context Pack вместо raw data access

Агент получает не базу данных, а ограниченный пакет смысла под конкретную задачу.

- Цель запроса
- Разрешённые объекты
- Термины и связи
- Агрегаты и расчёты
- Audit marker

Контекст должен быть purpose-bound: достаточный для задачи, минимальный по данным, проверяемый по происхождению.

LLM плохо считает. LLM хорошо интерпретирует.

Расчёты должны быть детерминированными, а не результатом генерации текста.

Детерминированный compute layer считает

KPI · SLA · балансы · OEE/utilization · производительность · отклонения · ETA · cost/throughput · риск и приоритет

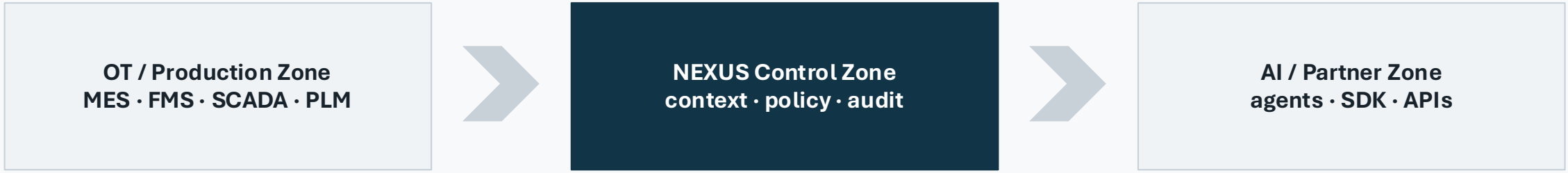
LLM-интерпретатор объясняет

что изменилось · почему это важно · какие есть гипотезы · что спросить у эксперта · какой следующий шаг в рамках политики

Нельзя путать ответственность за расчёт и ответственность за объяснение.

Безопасность NEXUS — это не “ограничить чат”

Это контроль зоны, цели, полномочий, данных, egress и последствий действия агента.



RBAC/ABAC + identity federation
policy engine, masking, redaction
partner isolation and purpose-bound access
immutable audit of context, prompt, response, tool calls
no direct agent access to production databases

Принцип: не давать агенту право, которое нельзя объяснить аудитору.

ISA/IEC 62443: zones/conduits; OWASP Agentic AI: privilege and tool misuse risks

Стек должен быть слоистым, заменяемым и on-prem ready

Для CIO важно не имя вендора, а контролируемая архитектура внедрения.

Identity / IAM	Keycloak · corporate IdP · OIDC/SAML
Policy / Security	OPA/Rego · ABAC/RBAC · DLP · egress control
Integration / Events	API Gateway · NATS JetStream / Kafka · webhooks
Data / Analytics	PostgreSQL · ClickHouse · object storage · historian connectors
Semantic / Retrieval	SKOS/OWL · property graph · vector search · GraphRAG
Compute / Rules	KPI services · rules engine · materialized views
Agent Interface	Context Pack API · Semantic Query API · SDK · MCP-compatible gateway
Observability	OpenTelemetry · Prometheus/Grafana · audit ledger

Важнее всего: контракты между слоями, а не конкретный бренд компонента.

Промышленный контекст меняется по отрасли. Паттерн NEXUS остаётся тем же.

Источники и термины разные, но слои “смысл → расчёт → доступ → аудит” повторяются.

Горная промышленность

FMS/MES/IIoT: смена, техника, карта, рейс, простой, материал, безопасность.

Машиностроение

PLM/Teamcenter: ТП, операция, ГОСТ, инструмент, режимы, материал, writeback.

Логистика и транспорт

TMS/WMS: заказ, маршрут, ETA, SLA, задержка, перевозчик, клиентская зона.

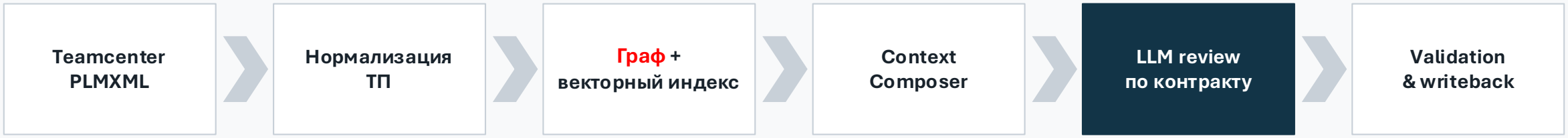
Строительство

BIM/ERP/PM: объект, WBS, график, подрядчик, техника, материалы, акты, риски.

NEXUS — это не отраслевой чат-бот. Это повторяемая архитектура промышленного AI-доступа.

Teamcenter + ЕСТД/ГОСТ + LLM: ценность не в модели, а в интеграционном слое

Рабочий пример NEXUS-подхода на домене технологических процессов механообработки.



PLMXML/TcXML приводится к канонической модели технологического процесса
графовая онтология связывает материалы, инструменты, операции, оборудование и стандарты
LLM получает только подготовленный context pack, без доступа к БД и файлам
ответ валидируется как JSON/Finding и отображается в отчёт или writeback-контур

Это и есть NEXUS-pattern: connector + ontology + context + contract + validation + audit.

ЧЕМ NEXUS ОТЛИЧАЕТСЯ ОТ ОБЫЧНОГО RAG

Обычный RAG ищет фрагменты. NEXUS управляет промышленным смыслом.

Для производственных доменов “похожий текст” недостаточен. Нужны связи, версии, правила и расчёты.

Naive RAG

поиск похожих фрагментов текста; слабая работа с версиями, связями, статусами и правами доступа

NEXUS / GraphRAG pattern

граф связей + векторный поиск + детерминированные проверки + context pack + audit trail

В промышленности вопрос не “нашли ли документ?”, а “поняли ли применимость, версию, связь и последствия?”.

Microsoft GraphRAG: graph + retrieval + LLM prompting/summarization

ФИНАЛЬНЫЙ ТЕЗИС

AI в промышленности начинается не с модели. Он начинается с архитектуры доверия.

Модель можно заменить. Потерянный контроль над данными, доступом и смыслом заменить намного сложнее.

NEXUS = Data Control + Terminology + Ontology + Context Packs + Deterministic Compute + Agent Interfaces + Security + Audit

Вопрос для аудитории:

“Готова ли наша инфраструктура объяснить, что именно знал AI, почему он имел право это знать, кто это посчитал и как это проверить?”

Если ответ “нет” — у компании ещё нет промышленного AI-доступа. Есть только эксперимент.