АНАЛИТИЧЕСКАЯ ЗАПИСКА

# ИИ-ассистент в диспетчерской: что он меняет и как внедрять безопасно

Большие языковые модели добрались до диспетчеризации зданий и технологических процессов. Записка объясняет, какие задачи службы эксплуатации ИИ-ассистент решает уже сегодня, как должно быть устроено безопасное решение, что показала демонстрация на действующем объекте — и по каким критериям оценивать такие продукты.

| KIS-WP-AI-2026-001 | 0.1 | 24.08.2026 | ~15 минут |
| --- | --- | --- | --- |
| Обозначение документа | Версия | Дата | Время чтения |

## 1. Резюме — пять тезисов

1. **Диспетчеризация упирается не в данные, а в человека.** Современные системы собирают всё; узкое место — разбор: сотни срабатываний за смену, экспертиза в головах старших смен, аналитика «по запросу к инженеру» вместо «по запросу словами».

2. **ИИ-ассистент — это не чат-бот со справкой, а инструмент поверх живых данных.** Решение имеет ценность, только если модель через инструментальный слой читает журналы, тренды и модель объекта конкретной системы диспетчеризации — и отвечает по ним, а не «в целом».

3. **Безопасность достигается построением, а не обещанием.** Работа строго в границах прав пользователя, отдельные разрешения на изменения, подтверждение каждого действия человеком, полный журнал — эти механизмы проверяемы службой ИБ.

4. **Размещение модели — выбор заказчика, а не ограничение технологии.** Локальная (on-premise) модель для КИИ и закрытых контуров, облачная — там, где допустимо; правильная архитектура поддерживает оба варианта.

5. **Технология уже работает.** Работающие ассистент и агент доступны в составе российской платформы диспетчеризации. Лучший способ убедиться — живая демонстрация на данных вашего объекта в режиме «только чтение».

## 2. Почему сейчас: три разрыва в работе диспетчерской

За последние двадцать лет диспетчеризация решила задачу сбора: современный объект отдаёт в систему тысячи и десятки тысяч сигналов, история пишется годами, тревоги настроены по каждой подсистеме. Но три разрыва остались — и все три упираются в человека.

### 2.1. Сигналов много — диагнозов мало

Журнал тревог фиксирует *события*, а служба живёт *причинами*. Типичная неделя крупного здания — сотни срабатываний, за которыми стоит несколько реальных первопричин: «дребезжащая» уставка, забитый фильтр, неисправный датчик. Превращение потока событий в диагноз — ручная работа: выборки по журналам, сопоставление трендов, опыт. На неё редко хватает смены, поэтому разбор происходит по остаточному принципу — либо после инцидента.

### 2.2. Диспетчеризация работает круглосуточно — опыт нет

Качество разбора нештатной ситуации ночью и в выходные зависит от того, кто на смене. Знания о «капризном» оборудовании, типовых ложных срабатываниях и порядке действий живут в головах и передаются устно; с уходом старшего смены они уходят тоже. Регламенты и паспорта существуют, но в момент аварии их никто не открывает.

### 2.3. Данные есть — аналитики по запросу нет

Ответ на вопрос руководителя «почему вырос расход воды?» или «что у нас чаще всего отказывает?» требует инженера, который сделает выборки, построит графики и интерпретирует их. Данные лежат в системе, но путь от вопроса обычными словами до ответа с цифрами занимает часы или дни — поэтому большинство таких вопросов просто не задаётся.

**СУТЬ МОМЕНТА**

Большие языковые модели (LLM — large language model) закрывают именно эти три разрыва: они умеют читать журналы и тренды как данные, объяснять их обычным языком и выполнять рутинные выборки и сопоставления за секунды. Вопрос сместился с «возможно ли это» на «как встроить это в диспетчерскую безопасно».

## 3. Что такое ИИ-ассистент диспетчерской — и чем он не является

**ИИ-ассистент диспетчерской** — это диалоговый интерфейс к работающей системе диспетчеризации: пользователь задаёт вопрос или поручение обычными словами, ассистент выполняет его, обращаясь к данным и функциям платформы, и отвечает с опорой на конкретные значения, журналы и документы объекта.

### 3.1. Три класса задач

| Класс | Примеры | Что требуется от решения |
|---|---|---|
| **Объяснение** | «Почему сработала авария в котельной?», «что менялось в проекте за неделю?», «кто квитировал?» | доступ к журналам, истории значений и модели объекта |
| **Аналитика по запросу** | сводка за смену, сравнение периодов, поиск корреляций и аномалий, отчёт таблицей или графиком | доступ к архиву; умение считать, а не «фантазировать» |
| **Действия агента** | создать заявку, набор тегов или тревог, черновик мнемосхемы, скрипт — по словесному описанию | инструментальный доступ к операциям платформы + контроль человека |

### 3.2. Два режима работы

Практика показывает, что у ассистента две разные аудитории с разными задачами — и хорошее решение явно разделяет эти режимы:

- **Режим оператора (диспетчера).** Объяснение тревог до первопричины, ответы по журналам, сводки и графики, подсказки по регламентам объекта. Для этого режима достаточно прав «только чтение».
- **Режим инженера.** Массовое создание тегов, тревог и трендов, черновики мнемосхем и логики, ревизия настроек проекта. Здесь ассистент действует — и потому каждое изменение показывается перед применением и протоколируется.

### 3.3. Чем ассистент не является

- **Не чат-бот с общими знаниями.** Модель, отвечающая «в целом про вентиляцию», для диспетчерской бесполезна: ценность создаёт только связка с живыми данными конкретного объекта.
- **Не «автопилот» объекта.** Ассистент не управляет оборудованием самостоятельно: агентские действия — это операции конфигурирования и организации работы, каждая из которых подтверждается человеком.
- **Не классическая предиктивная аналитика.** Математические модели прогноза отказов — отдельный класс инструментов; ассистент их дополняет (и делает их результаты доступными в диалоге), но не заменяет.

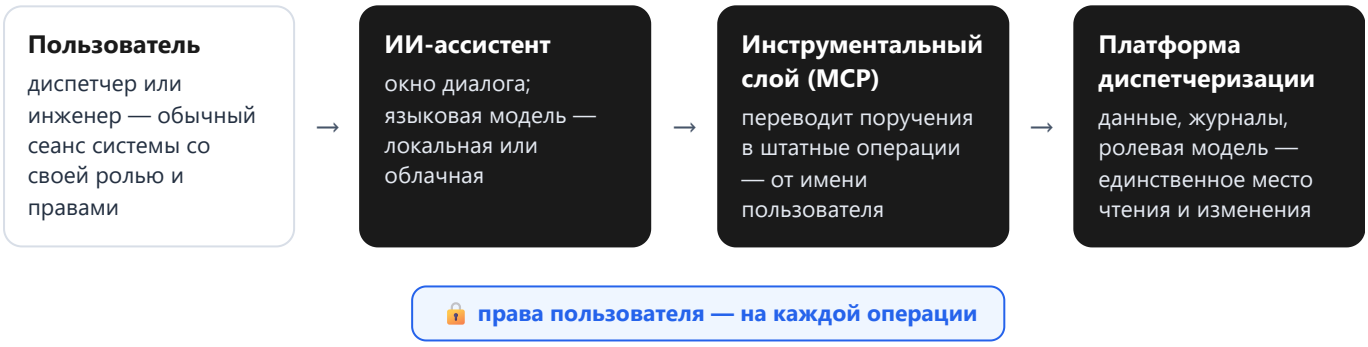
# 4. Как устроено правильное решение

## 4.1. Инструментальный слой — главное архитектурное решение

Языковая модель сама по себе не «видит» систему диспетчеризации. Связывает их **инструментальный слой** — служба, которая предоставляет модели набор именованных операций: прочитать значения, сделать выборку из журнала, найти объект в дереве, создать заявку. Индустриальным стандартом такого слоя становится открытый протокол MCP (Model Context Protocol). Принципиальные следствия:

- каждое обращение ассистента к данным — это **штатная операция платформы**, проходящая её проверки прав, а не прямой доступ к базе;
- ассистент **интерпретирует проект по его модели данных** (дерево объекта, имена, связи) — решение не требует обучения модели под каждый объект;
- список доступных операций конечен и известен — службе ИБ есть что аудировать.

## 4.2. Контур исполнения




## 4.3. База знаний объекта

Вторая опора решения — **база знаний**: регламенты, инструкции, паспорта оборудования и схемы объекта, проиндексированные для поиска. Благодаря ей ассистент отвечает не «как обычно устроены ИТП», а «как устроен ваш ИТП, по вашему регламенту» — со ссылкой на источник. База знаний пополняется в эксплуатации, и вместе с ней накапливается опыт службы — который больше не уходит вместе с людьми.

## 4.4. Безопасность: риск → механизм

| Риск, который видит служба ИБ | Механизм правильного решения |
|---|---|
| Ассистент получит доступ шире, чем пользователь | работа строго от имени текущей учётной записи, в границах её ролей; отдельный канал доступа не создаётся |
| ИИ «самовольно» изменит настройки или уставки | отдельные разрешения на изменения (ассистенту можно разрешить меньше, чем себе) + показ каждого изменения и явное подтверждение человеком |
| Действия ИИ нельзя будет разобрать постфактум | каждый шаг протоколируется штатным журналом: что сделано, когда, в чьём сеансе |
| Данные объекта уйдут во внешние сервисы | вариант локального (on-premise) размещения модели и базы знаний — внешних вызовов нет вовсе |
| Модель «выдумает» значения и цифры | ответы строятся на результатах инструментальных операций; источники (журнал, период, счётчик) называются в ответе и проверяемы |
| Нештатное поведение под нагрузкой или в аварии | ассистент — надстройка: его отказ не влияет на работу диспетчеризации; критические функции остаются за платформой и человеком |

### 4.5. Размещение модели: локально или в облаке

| Критерий | Локальная модель (on-premise) | Облачная модель |
|---|---|---|
| **Кому подходит** | КИИ, закрытые контуры, строгие политики ИБ | объекты без запрета внешних сервисов |
| **Данные** | не покидают периметр предприятия | запросы уходят провайдеру модели |
| **Инфраструктура** | требуется сервер с ускорителем под модель | дополнительное оборудование не требуется |
| **Качество моделей** | открытые модели; растёт быстро | доступны наиболее сильные модели |
| **Старт** | дольше (развёртывание контура) | быстрее |

Правильная архитектура поддерживает **оба варианта** — выбор делает служба ИБ заказчика, и его можно пересмотреть без замены решения: контур исполнения и база знаний не меняются.

# 5. Что это даёт службе — и где не ждать чуда

## 5.1. Эффект по ролям

| Роль | Эффект |
|---|---|
| **Смена** | разбор тревог до первопричины и ответы по журналам — одинаково глубокие в любую смену; быстрый ввод новых дежурных (ассистент подсказывает по регламентам объекта) |
| **Инженер** | сотни однотипных операций конфигурирования — одной командой; ревизия настроек вопросами вместо раскопок по проекту |
| **Руководитель** | отчёты, сравнения и оценки экономии — вопросом в чате, без ожидания «когда у инженера дойдут руки» |
| **Служба ИБ** | проверяемые механизмы контроля (раздел 4.4) вместо запрета технологии «на всякий случай» |

## 5.2. Границы применимости — честно

- **Качество данных определяет качество ответов.** Если датчики неисправны, а тревоги настроены формально, ассистент это обнаружит и назовёт (что само по себе ценно) — но аналитика будет ограничена. Первый эффект внедрения часто — «наведение порядка в учёте».
- **База знаний требует документов.** Ответы «по вашему регламенту» возможны, если регламенты существуют и загружены. Устаревшая документация даст устаревшие подсказки — нужен владелец базы знаний.
- **Это ассистент, а не замена службы.** Решения принимает человек; ассистент сокращает рутину и уравнивает смены, а не отменяет квалификацию.
- **Экономический эффект зависит от объекта.** Отраслевые диапазоны оптимизации — ориентиры; на вашем объекте цифры покажет только обследование.

## 6. Методика внедрения

1. **Платформа.** Ассистент — надстройка над системой диспетчеризации: он внедряется на действующем или новом проекте, где уже есть данные, журналы и модель объекта.
2. **База знаний.** Сбор и загрузка документации объекта: регламенты, паспорта, инструкции, схемы. Практический совет: начинать с документов, по которым чаще всего звонят «старшему».
3. **Права и сценарии.** Решение службы ИБ о размещении модели (локально/облако); настройка ролей и отдельных разрешений ассистента; согласование перечня агентских действий. Разумный старт — режим «только чтение» для всех.
4. **Эксплуатация.** Служба работает с ассистентом в обычном интерфейсе; база знаний пополняется по итогам инцидентов — качество ответов растёт со временем.

**РЕКОМЕНДУЕМЫЙ ПЕРВЫЙ ШАГ**

Демонстрация на данных вашего объекта в режиме «только чтение»: подключение к действующему проекту, сессия вопросов вашей службы, разбор результатов. Это быстрее и убедительнее любых слайдов — и не несёт рисков для эксплуатации.

## 7. Чек-лист: 10 вопросов к любому ИИ-решению для диспетчерской

Вопросы ниже применимы к любому вендору; отрицательный ответ на первые четыре обесценивает остальное.

| № | Вопрос | Что считать хорошим ответом |
|---|---|---|
| 1 | Работает ли ассистент с живыми данными системы (значения, журналы, тренды) — или только со справкой? | инструментальный доступ к платформе; ответы по данным объекта |
| 2 | Показать работу на действующем объекте, а не на слайдах? | живая демонстрация, лучше — на ваших данных |
| 3 | В чьих правах действует ассистент? | строго в границах прав текущего пользователя; отдельного «служебного» суперпользователя нет |
| 4 | Может ли ассистент что-то изменить без подтверждения человека? | нет; каждое изменение показывается и подтверждается |
| 5 | Протоколируются ли действия ИИ? | каждый шаг — в штатном журнале системы |
| 6 | Возможно ли полностью локальное развёртывание (модель и база знаний)? | да, для КИИ и закрытых контуров; облачный вариант — по выбору |
| 7 | Отвечает ли ассистент по документации именно нашего объекта? | база знаний объекта с индексированием; ответы со ссылкой на источник |
| 8 | Нужно ли обучать/дообучать модель под наш объект? | нет; ассистент интерпретирует проект по его модели данных |
| 9 | Что происходит с диспетчеризацией при отказе ассистента? | ничего: ассистент — надстройка, не входящая в контур управления |
| 10 | Работает ли решение на русском языке — интерфейс, диалог, документация? | да, полностью |

## 8. Как это реализовано в Платформе ГАУС

Описанная в записке архитектура — не теоретическая модель: именно так устроен **ГАУС.Copilot**, ИИ-ассистент и активный агент российской платформы диспетчеризации ГАУС (запись в реестре российского ПО №11422). Кратко:

- ассистент встроен в клиент платформы; инструментальный слой (MCP-сервер) входит в поставку;
- два режима — диспетчера и инженера; агентские действия с подтверждением и протоколированием;
- языковая модель — локальная (on-premise) или облачная, по выбору службы ИБ заказчика;
- база знаний объекта — регламенты, паспорта, инструкции с ответами со ссылкой на источник;

Подробнее: датшит платформы (KIS-DS-PLT-2026-001), one-pager «ГАУС.Copilot» (KIS-OP-AI-2026-001), презентация «ГАУС.Copilot» (KIS-PR-AI-2026-001).

**8 800 600 15 87**

sales@info-systems.pro · ООО «Комплексные Инфосистемы», группа компаний ТехноИнжиниринг

**Демо на вашем объекте**

режим «только чтение», сессия вопросов вашей службы

## 9. Глоссарий

| Термин | Значение |
|---|---|
| **LLM (большая языковая модель)** | класс моделей ИИ, понимающих и порождающих текст; основа диалоговых ассистентов |
| **ИИ-агент** | режим, в котором модель не только отвечает, но и выполняет действия через предоставленные ей инструменты |
| **MCP (Model Context Protocol)** | открытый протокол подключения инструментов и данных к языковым моделям; используется как инструментальный слой |
| **On-premise** | размещение программного обеспечения (здесь — языковой модели) на серверах предприятия, без внешних сервисов |
| **КИИ** | критическая информационная инфраструктура (187-ФЗ); объекты с повышенными требованиями к защите информации |
| **Квитирование** | подтверждение оператором факта ознакомления с тревогой |

Документ носит справочный характер и не является офертой. Цифры демонстрации — раздел 5 со сноской-методикой; утверждение о статусе на рынке — по анализу открытых материалов вендоров (август 2026). Воспроизведение с указанием источника приветствуется.