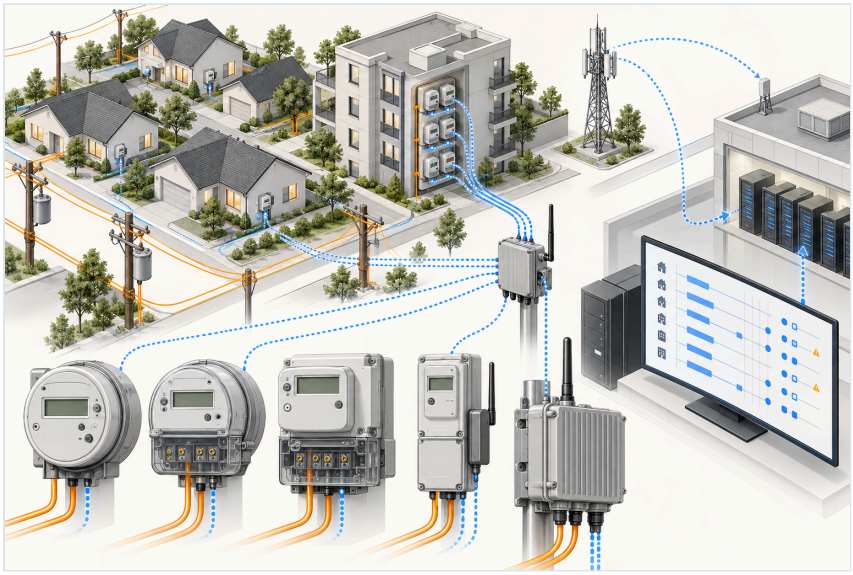
AETHERONE / AETHERHES

Техническое описание AetherHES

Head End System (HES) для связи с устройствами, сбора данных и выполнения команд · Редакция документа 2026.10



Концептуальная иллюстрация тракта сбора данных AMI: интеллектуальные счётчики в жилых домах и в шкафу учёта многоквартирного дома, счётчики разных конструкций, концентраторы данных на опорах с антеннами и вышка сотовой связи; синие линии связи ведут к серверным стойкам головной системы и к монитору сбора данных. Иллюстрация показывает тракт, который обслуживает AetherHES; это не установка заказчика и не запись о фактической работе.

Краткая сводка для выбора

AetherHES — головная система (head end) AetherONE: продукт, который ведёт обмен данными с интеллектуальными счётчиками, концентраторами данных (DCU) и шлюзами. Для каждого устройства он хранит коммуникационную идентичность и сеанс протокола, выполняет считывание и сбор данных, фиксирует, что произошло с каждой командой, отслеживает телеметрию каналов связи и события устройств, доставляет микропрограммы и конфигурацию, а необработанную запись обмена (Raw Chronicle) хранит в собственном закрытом хранилище. За пределы HES уходит типизированное декодированное событие измерения со ссылками и хешами. Необработанные байты остаются внутри.

В таблице ниже приведён утверждённый объём AetherHES и по каждой строке указано, что эта редакция предлагает для оценки.

Кому адресован документ

- **Инженерам по интеграции**, которые прослеживают показание от кадра устройства до декодированного события и подтверждения MDMS.
- **Инженерам эксплуатации**, которым нужно отличать подтверждение устройства от совпавшего контрольного считывания, а контрольное считывание — от итогового бизнес-результата.
- **Операторам сетей связи и парка устройств**, которые оценивают представления парка, каналов связи и микропрограмм.
- **Архитекторам решений**, которым важно видеть, где заканчивается HES и где начинаются MDMS, GIS, Revenue и Dispatch.

Возможности

Возможность	Что делает AetherHES	В этой редакции
Шлюз адаптера и сеансов	Принимает кадры устройств через локальный TCP-адаптер в формате JSON с разделением строк (newline-delimited JSON). Хранит каждый сеанс как открытый или закрытый, привязанный к своему устройству, в принадлежащем HES хранилище SQLite WAL и возобновляет работу после перезапуска.	Доступно для оценки как локальный компонент на одном хосте.
Сбор данных и Raw Chronicle	Записывает каждый полученный кадр в транзакционный журнал необработанных данных с надёжным outbox. Формирует версионированное декодированное событие измерения с вычисленными хешем исходного кадра и хешем полезной нагрузки, а также со ссылками. Публикует событие через отдельный спул брокера с доставкой, безопасной при повторах, и сохраняемым подтверждением.	Доступно для оценки на одном хосте. Декодированная полезная нагрузка — ограниченное показание в kW, а не общая модель OBIS или профиля нагрузки.
Живое представление только для чтения	Отдаёт страницы Fleet (парк устройств) и Collection (сбор данных) через loopback-API, который принимает только GET-запросы и применяет на стороне сервера список разрешённых полей. Детерминированные демонстрационные данные появляются лишь по явному запросу и никогда не подменяют неудавшееся живое считывание.	Локальное представление для оценки, а не сервис промышленной эксплуатации с аутентификацией.
Command Execution Queue	Резидентный обработчик в единственном экземпляре (pump) читает запросы из одного спула и добавляет упорядоченные свидетельства о статусе в другой. Он идемпотентно обрабатывает каждый запрос, привязывает цель к наблюдаемому сеансу и фиксирует подтверждение транспорта, подтверждение устройства и совпавшее контрольное считывание.	Доступно для оценки как поток доставки токена в тестовом режиме на детерминированном адаптере, только с локальными свидетельствами.
Обзор операций, сеть, микропрограммы, события и аварийные сигналы	Панель, топология и качество каналов связи, репозиторий микропрограмм, кампании и дрейф конфигурации, поток событий, входящие аварийные сигналы и просмотр необработанных кадров.	Демонстрационные интерфейсы на синтетических данных, показывающие предполагаемую картину эксплуатации.

Как оценивается команда

Команда проходит восемь стадий: запрошена, авторизована, поставлена в очередь, отправлена, подтверждена транспортом, подтверждена устройством, проверена контрольным считыванием и подтверждён бизнес-результат. Вся цепочка строится на различии между шестой и восьмой стадиями. Устройство, подтвердившее команду, не обязательно изменило состояние, а изменившееся состояние ещё не есть бизнес-результат, поэтому подтверждение устройства никогда не считается конечным состоянием. В тестовом потоке токена HES сообщает свидетельства выполнения, а завершённый бизнес-результат проецирует только AetherMDMS — после совпадения контрольного считывания.

Интеграция и ответственность за запись в закрытое хранилище

HES пишет только в собственное хранилище и не читает и не изменяет закрытую базу данных AetherMDMS. Всё, что пересекает границу между продуктами, передаётся как версионированный контракт или проекция только для чтения.

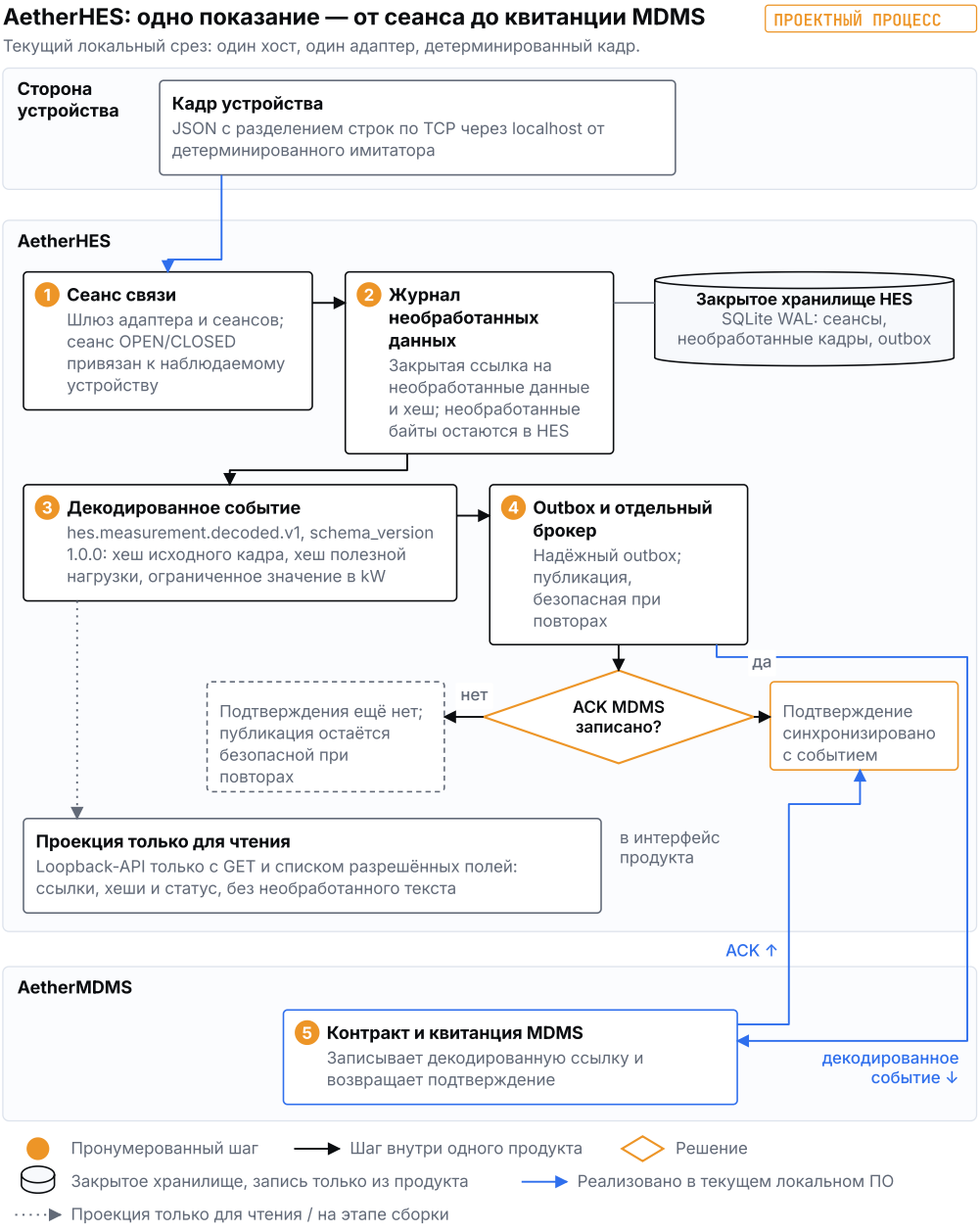
Смежный продукт	Что пересекает границу
AetherMDMS	Декодированные события измерения через hes.measurement.decoded.v1: типизированные поля, в том числе ограниченное значение в kW, а также ссылки и хеши. При обмене токенами в тестовом режиме MDMS отправляет mdms.token.delivery-requested.v1, а HES отвечает hes.token.delivery-status-changed.v1. Текст необработанных кадров, закрытый JSON событий, JSON подтверждений и материал токенов остаются внутри HES и не попадают ни в проекции для браузера, ни в MDMS.
AetherGIS	Отвечает за пространственные данные как первоисточник. Представления сети и топологии в HES — его проекции.
AetherRevenue	Отвечает за финансовое выставление счетов и расчёты. HES сообщает только свидетельства выполнения и не имеет финансовых полномочий.
AetherDispatch	Отвечает за полевые работы коммунального предприятия. Command Execution Queue — исполнительный контур HES, обращённый к устройствам, а не полевая диспетчеризация.
AetherUtilityConsole	Показывает HES в едином фронтенде через навигацию и композицию только для чтения. Общим бэкендом не является.

Редакция и условия оценки

У AetherHES есть утверждённый проектный объём, и эта редакция делает часть его доступной для контролируемой локальной оценки. Поставляемые компоненты рассчитаны на один хост и детерминированный имитируемый адаптер: шлюз сеансов, Raw Chronicle с декодированными событиями и спулом брокера, представления Fleet и Collection только для чтения и командный поток в тестовом режиме. Представления центра операций — демонстрационные интерфейсы, показывающие предполагаемую картину эксплуатации. За рамками этой редакции остаются промышленные пакеты протоколов DLMS, IDIS и SMGW, вендорская сертификация, аутентификация и утверждение операторов, высокая доступность и интеграция в реальных полевых условиях; командный поток — это доставка токена в тестовом режиме, а не промышленный STS. Показатели пропускной способности, задержки и ёмкости не публикуются, а Raw Chronicle — транзакционное локальное хранилище, которое не представляется защищённым от подделки.

Технические характеристики

- **Контракты.** hes.measurement.decoded.v1 (schema_version 1.0.0); mdms.token.delivery-requested.v1; hes.token.delivery-status-changed.v1.
- **Пакеты и среда выполнения.** Веб-пакет закрытый, версия 0.1.0, коммерческим релизом не является. Веб-интерфейс собирается на Node.js 20.19 или новее в линейке 20 либо на 22.12 или новее. Служба измерений требует Node.js 24 или новее.
- **Обработчик команд (pump).** Для его включения нужны явно заданный общий путь брокера и парные настройки спулов запросов и статусов токенов.
- **Редакция документа.** 2026.10 — обозначение этого документа, а не релиз продукта.



ПРОЕКТНЫЙ ПРОЦЕСС Проектный процесс: сеанс связи, закрытая ссылка на необработанные данные и их хеш, декодированное событие, отдельный брокер, контракт MDMS и квитанция. Это концептуальная проектная диаграмма предполагаемого пути одного показания; она не описывает интеграцию в промышленной эксплуатации.