AETHERONE / AETHERHES

Ficha técnica de AetherHES

Sistema head-end (HES) para la comunicación con dispositivos, la recopilación de datos y la ejecución de comandos · Edición del documento 2026.10

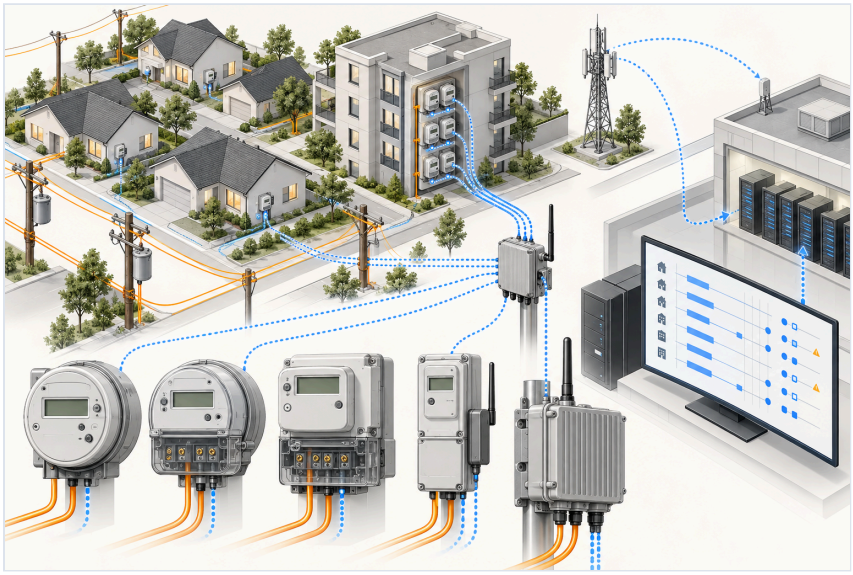


Ilustración conceptual de una ruta de adquisición AMI: contadores inteligentes en viviendas y en el armario de contadores de un bloque de apartamentos, contadores de varios diseños, concentradores de datos instalados en postes con antenas y una torre de telefonía móvil, con trayectos de comunicación azules que llegan a bastidores de servidores head-end y a un monitor de recopilación. Muestra la ruta a la que sirve AetherHES y no es una instalación de cliente ni un registro de operación.

Resumen para la selección

AetherHES es el head-end de AetherONE: el producto que mantiene la comunicación con los contadores inteligentes, los concentradores de datos (DCU) y las pasarelas. Para cada dispositivo conserva una identidad de comunicación y una sesión de protocolo, ejecuta lecturas y recopilación, registra lo que ocurrió con cada comando, sigue la telemetría del enlace y los eventos del dispositivo, entrega firmware y configuración y guarda el registro de comunicación sin procesar (Raw Chronicle) en un almacén privado. Lo que sale de HES es un evento de medición decodificado y tipado, con referencias y hashes. Los bytes sin procesar se quedan dentro.

La tabla siguiente recoge el alcance aprobado de AetherHES e indica, fila por fila, qué ofrece esta edición para su evaluación.

A quién va dirigida

- **Ingenieros de integración**, que siguen una lectura desde la trama del dispositivo hasta un evento decodificado y un acuse de MDMS.
- **Ingenieros de operaciones**, que deben distinguir el acuse de un dispositivo de una lectura de comprobación coincidente, y esta de un resultado de negocio final.
- **Operadores de redes de comunicación y de flotas de dispositivos**, que evalúan las vistas de flota, de enlaces y de firmware.
- **Arquitectos de soluciones**, que necesitan ver dónde termina HES y dónde empiezan MDMS, GIS, Revenue y Dispatch.

Capacidades

Capacidad	Qué hace AetherHES	En esta edición
Pasarela de adaptador y sesiones	Acepta tramas de dispositivos a través de un adaptador TCP local que usa JSON delimitado por saltos de línea. Mantiene cada sesión abierta o cerrada, vinculada a su dispositivo, en un almacén SQLite WAL propio de HES, y la retoma tras un reinicio.	Disponible para evaluación como componente local de un único host.
Recopilación y Raw Chronicle	Escribe cada trama recibida en un registro transaccional sin procesar con un outbox duradero. Genera un evento de medición decodificado y versionado, con un hash calculado de la trama sin procesar, un hash de la carga útil y referencias. Publica a través de un spool de broker independiente, con entrega segura ante repeticiones y acuse persistido.	Disponible para evaluación en un único host. La carga útil decodificada es una lectura de kW acotada, no un modelo general de OBIS ni de perfil de carga.
Vista en vivo de solo lectura	Sirve las páginas de flota (Fleet) y de recopilación (Collection) desde una API de loopback que solo admite GET, con una lista de campos permitidos aplicada en el servidor. Los datos de demostración deterministas aparecen únicamente si se piden de forma explícita y nunca sustituyen una lectura en vivo fallida.	Vista de evaluación local, no un servicio de producción con autenticación.
Command Execution Queue	Un proceso residente de instancia única (pump) lee las solicitudes de un spool y añade evidencia de estado ordenada a otro. Consume cada solicitud de forma idempotente, vincula el destino a la sesión observada y registra el acuse de transporte, el acuse del dispositivo y la lectura de comprobación coincidente.	Disponible para evaluación como flujo de entrega de tokens en modo de prueba sobre un adaptador determinista, solo con evidencia local.
Visión general de operaciones, red, firmware, eventos y alarmas	Panel, topología y calidad del enlace, repositorio de firmware, campañas y deriva de configuración, flujo de eventos, bandeja de alarmas y visor de tramas sin procesar.	Interfaces de demostración con datos sintéticos que muestran el panorama operativo previsto.

Cómo se evalúa un comando

Un comando recorre ocho etapas: solicitado, autorizado, en cola, enviado, con acuse de transporte, con acuse del dispositivo, verificado por lectura de comprobación y resultado de negocio confirmado. Toda la cadena gira en torno a la distancia entre la sexta etapa y la octava. Un dispositivo que ha acusado recibo de un comando no necesariamente ha cambiado de estado, y un estado cambiado todavía no es un resultado de negocio; por eso el acuse del dispositivo nunca es un estado final. En el flujo de tokens en modo de prueba, HES informa de la evidencia de ejecución y solo AetherMDMS proyecta un resultado de negocio completado, una vez que la lectura de comprobación coincide.

Integración y responsabilidad de escritura privada

HES escribe únicamente en su propio almacén y no lee ni escribe la base de datos privada de AetherMDMS. Todo lo que cruza el límite entre productos viaja como contrato versionado o como proyección de solo lectura.

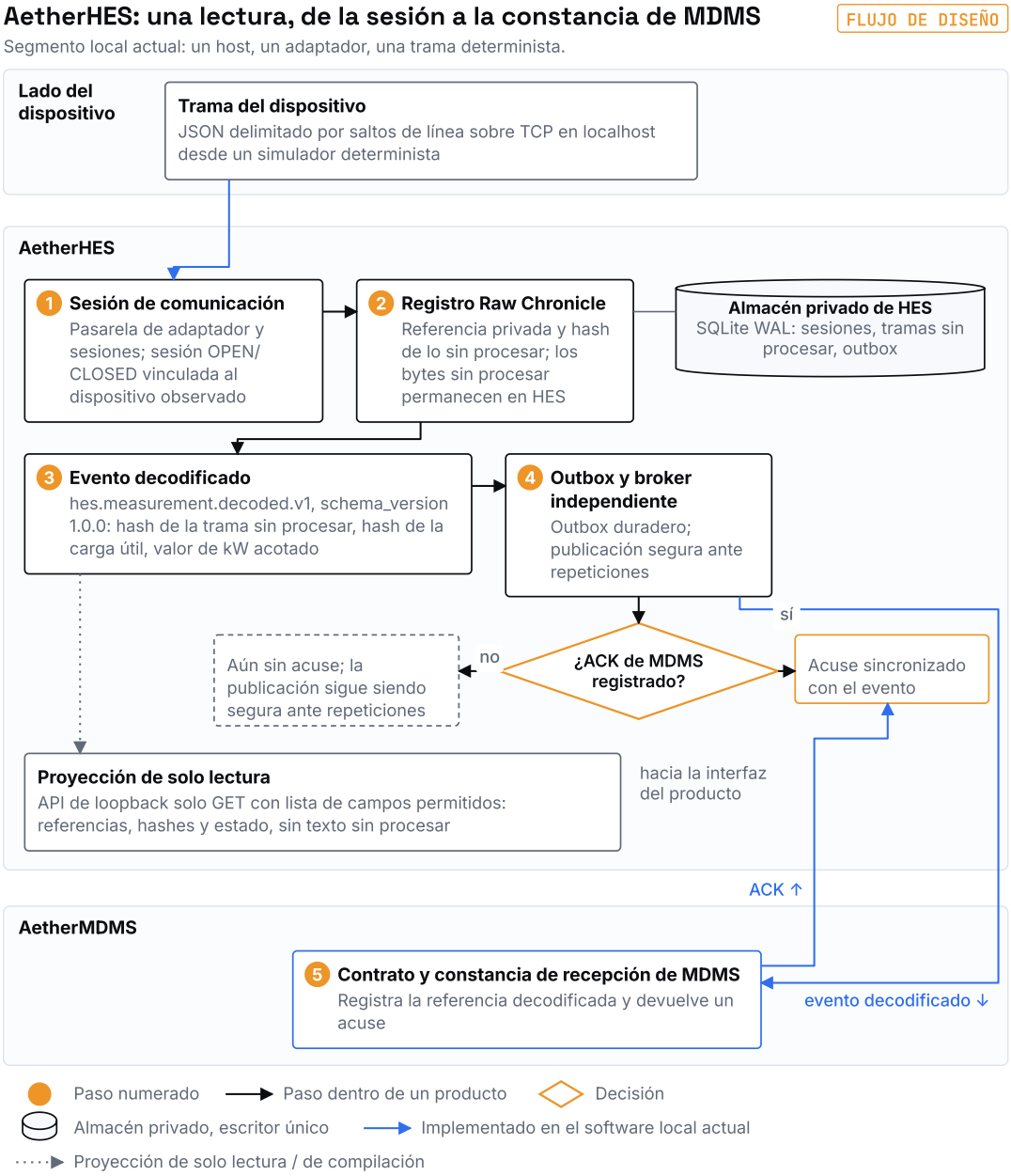
Contraparte	Qué cruza el límite
AetherMDMS	Eventos de medición decodificados mediante hes.measurement.decoded.v1: campos tipados, entre ellos un valor de kW acotado, además de referencias y hashes. En el intercambio de tokens en modo de prueba, MDMS envía mdms.token.delivery-requested.v1 y HES responde con hes.token.delivery-status-changed.v1. El texto de las tramas sin procesar, el JSON privado de eventos, el JSON de acuses y el material de los tokens permanecen dentro de HES y no aparecen ni en las proyecciones destinadas al navegador ni en MDMS.
AetherGIS	Es la fuente de verdad espacial. Las vistas de red y de topología de HES son proyecciones de ella.
AetherRevenue	Tiene a su cargo la facturación financiera y la liquidación. HES informa solo de evidencia de ejecución y no ejerce ninguna autoridad financiera.
AetherDispatch	Tiene a su cargo el trabajo de campo de la empresa de servicios públicos. La Command Execution Queue es la superficie de ejecución de HES orientada al dispositivo, no el despacho de campo.
AetherUtilityConsole	Presenta HES dentro del frontend unificado mediante navegación y composición de solo lectura. No es un backend compartido.

Condiciones de la edición y de la evaluación

AetherHES tiene un alcance de diseño aprobado, y esta edición pone una parte a disposición para una evaluación local controlada. Los componentes entregados están pensados para un único host y un adaptador simulado determinista: la pasarela de sesiones, el Raw Chronicle con eventos decodificados y un spool de broker, las vistas de flota y de recopilación de solo lectura y un flujo de comandos en modo de prueba. Las vistas del centro de operaciones son interfaces de demostración que muestran el panorama operativo previsto. Quedan fuera de esta edición los paquetes de protocolo DLMS, IDIS y SMGW de producción, la certificación de proveedores, la autenticación y la aprobación de operadores, la alta disponibilidad y la integración real en campo; además, el flujo de comandos es una entrega de tokens en modo de prueba, no un STS de producción. No se publica ninguna cifra de rendimiento, latencia ni capacidad, y el Raw Chronicle es un almacén local transaccional que no se presenta como a prueba de manipulaciones.

Especificaciones técnicas

- **Contratos.** hes.measurement.decoded.v1 (schema_version 1.0.0); mdms.token.delivery-requested.v1; hes.token.delivery-status-changed.v1.
- **Paquetes y entorno de ejecución.** El paquete web es privado, tiene la versión 0.1.0 y no es un lanzamiento comercial. La interfaz web se compila con Node.js 20.19 o posterior dentro de la línea 20, o con 22.12 o posterior. El servicio de medición requiere Node.js 24 o posterior.
- **Pump de comandos.** Para activarlo hace falta una ruta de broker compartida explícita y la configuración emparejada de los spools de solicitudes de token y de estado.
- **Edición del documento.** 2026.10 identifica este documento. No es un lanzamiento de producto.



FLUJO DE DISEÑO Flujo de diseño: sesión de comunicación, referencia privada y hash de la trama sin procesar, evento decodificado, broker independiente, contrato de MDMS y constancia de recepción. Es un diagrama conceptual de diseño de la ruta prevista de una lectura y no describe una integración en producción.