



# Home Assistant

## Systemdokumentation der Smart-Home-Installation

Joachim Berger

Stand: 17.07.2026

Obeco GmbH · Arenberger Str. 230A · 56077 Koblenz

Telefon +49 (0)261 963 44660 · [info@obecosupport.de](mailto:info@obecosupport.de)



# Home Assistant

Zusammenfassende Darstellung der Home-Assistant-Installation: Architektur, Funkinfrastruktur, angebundene Geräte, Betriebsführung und gelöste technische Problemstellungen. Die Notiz verdichtet die Detaildokumentation aus 20 Software/Home Assistant auf die Ebene, die für das Verständnis des Gesamtsystems nötig ist; für Konfigurationsdetails und Kommandos wird auf die jeweilige Einzelnotiz verwiesen.

Abgrenzung: Batteriespeicher (Marstek/Omnibattery) und Photovoltaik werden separat dokumentiert und sind hier bewusst nicht enthalten – siehe „Systemgrenzen“.

## Das System auf einen Blick

| Merkmal           | Ausprägung                                                                    |
|-------------------|-------------------------------------------------------------------------------|
| Plattform         | Home Assistant OS, virtualisiert als VM auf einem Proxmox-Server              |
| Rolle             | Zentrale Steuer- und Überwachungsebene für die Haustechnik                    |
| Funkinfrastruktur | Zigbee 3.0 über eigenen Coordinator, angebunden via Zigbee2MQTT + MQTT-Broker |
| Weitere Anbindung | WLAN-Geräte (Shelly), IP-basierte Messtechnik, Cloud-APIs                     |
| Bedienung         | Web-Dashboard, Wandterminal im Kiosk-Modus, Push auf Mobilgerät               |
| Datenhaltung      | Recorder-Datenbank (SQLite), Aufbewahrung 365 Tage                            |
| Last (gemessen)   | ~38.750 Zustandsänderungen/Stunde, ~26 Schreib-IOPS                           |

Das System integriert Geräte unterschiedlicher Hersteller und Funkstandards zu einer einheitlichen Steuer- und Auswertungsebene. Kennzeichnend ist eine konsequent lokale Betriebsweise: Sensorik und Automationen funktionieren ohne Cloud-Anbindung; externe Dienste werden nur dort genutzt, wo sie inhaltlich erforderlich sind (z. B. Strompreise).

## Architektur

### Virtualisierung und Grundaufbau

Home Assistant läuft als eigene virtuelle Maschine (Home Assistant OS) auf einem Proxmox-Hypervisor. Diese Trennung erlaubt Snapshots vor Eingriffen, eine vom Hostsystem unabhängige Wartung und den parallelen Betrieb weiterer Dienste auf derselben Hardware. Die Systemdisk liegt auf einem NVMe-Speicherpool.

Ein zweiter Dienst – Zigbee2MQTT – läuft nicht innerhalb der HA-VM, sondern als eigener LXC-Container direkt auf dem Hypervisor. Diese Entkopplung ist eine bewusste Architekturentscheidung (siehe unten).



## Datenwege

Die Anbindung der Geräte erfolgt über drei Wege:

| Weg       | Technik                                                    | Beispiele                  |
|-----------|------------------------------------------------------------|----------------------------|
| Zigbee    | Gerät → Coordinator (USB) → Zigbee2MQTT → MQTT-Broker → HA | Wasserventil, Sensorik     |
| WLAN/IP   | Gerät → LAN → Hersteller-Integration in HA                 | Shelly-Geräte, Messadapter |
| Cloud-API | Onlinedienst → Integration → HA                            | Strompreis-Daten           |

Das Netzwerk stellt eine FRITZ!Box bereit (WLAN, Ethernet, Internet). Der Zigbee-Funk ist davon unabhängig und benötigt keine Internetverbindung.

## Netzwerk- und Sicherheitsprinzipien

- Lokale Verarbeitung: Automationen laufen vollständig im Haus; ein Internetausfall legt die Steuerung nicht still.
- Zigbee-Verschlüsselung: AES-128 (Standard des Protokolls).
- Kontrolliertes Anlernen: Das Join-Fenster wird nach dem Pairing wieder geschlossen, damit keine fremden Geräte beitreten können.

## Zigbee-Infrastruktur

### Coordinator

Als Coordinator dient ein SONOFF Zigbee/Thread PoE Dongle Max (Chipsatz EFR32MG21, Zigbee 3.0). Er bildet das Funknetz, verwaltet Topologie und Routing und ist zentral im Gebäude platziert. Der Dongle hängt physisch am Proxmox-Host und wird per USB-Passthrough an den Zigbee2MQTT-Container durchgereicht.

Das Zigbee-Netz arbeitet als Mesh: Netzbetriebene Geräte leiten Signale weiter, wodurch Reichweite und Ausfallsicherheit mit jedem zusätzlichen Router-Gerät steigen. Batteriebetriebene Sensoren nehmen als End Devices am Netz teil.

### Architekturentscheidung: Zigbee2MQTT statt ZHA

Ursprünglich lief das Zigbee-Netz über die in Home Assistant eingebaute ZHA-Integration. Der Umstieg auf Zigbee2MQTT erfolgte bewusst und wurde als Migration durchgeführt.

Begründung:

| Kriterium            | Vorteil von Zigbee2MQTT                                                |
|----------------------|------------------------------------------------------------------------|
| Gerätekompatibilität | Unterstützt mit über 4.000 Modellen praktisch den gesamten Markt       |
| Unabhängigkeit       | Läuft außerhalb von HA weiter – HA-Updates berühren das Funknetz nicht |



| Kriterium            | Vorteil von Zigbee2MQTT                                        |
|----------------------|----------------------------------------------------------------|
| Diagnostizierbarkeit | Detailliertes Logging, gezieltes Debugging einzelner Geräte    |
| Offener Standard     | MQTT ist herstellerneutral und auch für andere Systeme nutzbar |

Bewusst in Kauf genommen: höhere Komponentenzahl (Dongle, Z2M, Broker, Integration), dadurch mehr Einrichtungs- und Fehlersuchaufwand sowie zusätzlicher Ressourcenbedarf.

## Migration und heutiger Betrieb

Die Ablösung von ZHA erfolgte im Mai 2026. Da Zigbee-Geräte an ihr Netz gebunden sind, mussten alle 10 Geräte neu angelernt werden; die Entitätsnamen änderten sich dabei (zha.\* → sensor.\*, switch.\* etc.), sodass Automationen, Skripte und Dashboards nachgezogen werden mussten. Als Rückfallebene diente ein Snapshot der HA-VM.

Der erste Umstieg nutzte Zigbee2MQTT noch als Add-on innerhalb der HA-VM. Der heutige produktive Aufbau betreibt Zigbee2MQTT als eigenständigen LXC-Container auf dem Hypervisor – konsequenter im Sinne der oben genannten Entkopplung. Die Anbindung an Home Assistant erfolgt über den MQTT-Broker; neue Geräte werden per Auto-Discovery automatisch als Entitäten angelegt.

## Angebundene Geräte und Anwendungsfälle

### Sicherheitstechnik

| Gerät                 | Funktion                                                                                                                                                                                                                               |
|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Shelly Gas (SHGS-1)   | Erkennung brennbarer Gase (Methan, Butan, Propan, Wasserstoff) über Pellistor-Sensor; Messung in % der unteren Explosionsgrenze (LEL), Alarmschwelle konfigurierbar. Arbeitet auch ohne Cloud, mit lokalem akustischem Signal und LED. |
| Shelly Flood (SHWT-1) | Wasserleck-Erkennung über Kontaktflächen, zusätzlich Temperaturmessung. Batteriebetrieb (CR123A, ca. 18 Monate), IP67. Vorgesehen für Keller, Heizungsraum und Nassbereiche.                                                           |

Bewusste Abgrenzung: Die Gaserkennung ist als Smart-Home-Sensorik für Automatisierung ausgelegt und ersetzt keine normkonformen Gasmelder nach DIN EN 50194. Konventionelle Melder sind zusätzlich vorzusehen.

### Verbrauchserfassung

Der Gasverbrauch wird über einen optischen Lesekopf am vorhandenen Zähler erfasst. Das Verfahren ist kontaktlos und nicht-invasiv – der Zähler selbst bleibt unverändert, es ist kein Eingriff durch den Netzbetreiber nötig. Die LED-Pulse werden gezählt (ein Puls entspricht 0,001 m<sup>3</sup>) und per Netzwerk an



Home Assistant übertragen.

## Bewässerungssteuerung

Ein SONOFF SWV-ZFE (Zigbee-Wasserventil mit integriertem Durchflusssensor) steuert die Bewässerung. Home Assistant erhält daraus Ventilzustand, Wasserverbrauch in Litern und Batteriestand. Der Betrieb erfolgt zeitgesteuert (täglich 06:00–21:00 Uhr) und wird durch eine Watchdog-Automation abgesichert – siehe „Gelöste Problemstellungen“.

## Bedienung: Wandterminal

Ein Shelly Wall Display dient als fest installiertes Bedienpanel. Da das Gerät keinen Play Store mitbringt, wurde der Fully Kiosk Browser per ADB-Sideload installiert und als Standard-Launcher gesetzt. Damit startet das Terminal nach einem Neustart selbsttätig im Kiosk-Modus und zeigt dauerhaft ein HA-Dashboard.

Berücksichtigte Randbedingungen: Die Auflösung von 480 × 480 px erfordert ein eigens dafür gebautes Dashboard mit großen Touchflächen; die Anmeldung erfolgt über ein langlebiges Zugriffstoken; die kamerabasierte Bewegungserkennung des Browsers arbeitet auf diesem Gerät unzuverlässig, weshalb das Aufwecken des Bildschirms stattdessen über einen PIR-Sensor und die REST-Schnittstelle des Browsers vorgesehen ist.

## Betrieb und Überwachung

### Zentrales Benachrichtigungs-Logbuch

Alle ausgehenden Benachrichtigungen – unabhängig davon, welche Automation sie auslöst – werden an einer Stelle protokolliert. Erfasst werden derzeit zwei Kanäle: Push auf das Mobilgerät und WhatsApp.

Technisch horcht eine Automation auf die Dienstaufrufe beider Kanäle und schreibt Kanal, Titel und Text ins Logbuch. Der Nutzen: Es ist jederzeit nachvollziehbar, was das System wann gemeldet hat – ohne die einzelnen Automationen durchsuchen zu müssen. Die Aufbewahrung ist auf die Recorder-Vorhaltung (365 Tage) abgestimmt; die Anzeige umfasst sechs Monate. Neue Kanäle lassen sich durch einen zusätzlichen Trigger ergänzen.

## Gelöste Problemstellungen

Die folgenden drei Fälle dokumentieren typische Fragestellungen im Betrieb einer heterogenen Smart-Home-Installation – jeweils mit Ursachenanalyse, Lösung und benannten Grenzen der Lösung.

### 1. Kostenanzeige um Faktor 100 verfälscht

Symptom: Das Energie-Dashboard wies die Netzkosten hundertfach zu hoch aus – 4,29 kWh wurden mit 144,65 € statt rund 1,45 € bewertet. Die Gaskosten waren korrekt.



**Ursache:** Home Assistant interpretiert eine Preis-Entität immer als Systemwährung pro kWh (hier EUR/kWh) – unabhängig davon, welche Einheit die Entität tatsächlich führt. Der verwendete Strompreis-Sensor liefert jedoch ct/kWh. Die Multiplikation „kWh × Zahlenwert“ ergibt damit einen um den Faktor 100 falschen Betrag. Es handelt sich nicht um einen Fehler der eigenen Konfiguration, sondern um bekanntes Verhalten von Home Assistant, das in den Projekt-Repositories dokumentiert und zum Stand Juli 2026 ungelöst ist.

**Lösung:** Ein Template-Helfer rechnet den Preis auf EUR/kWh um; im Energie-Dashboard wurde die Preisquelle auf diesen Helfer umgestellt.

**Grenze der Lösung:** Der Fix wirkt ausschließlich auf neue Daten. Home Assistant berechnet die Kosten stündlich und schreibt sie fest; bereits gespeicherte Fehlwerte werden nicht rückwirkend korrigiert. Zurückliegende Tage bleiben verfälscht, solange sie nicht manuell über die Statistik-Korrektur bereinigt werden.

## 2. Wasserventil schaltet selbsttätig ab

**Symptom:** Das Zigbee-Ventil schaltete nach rund 10 Minuten selbsttätig ab – obwohl die Schutzfunktion „Abschalten bei Wassermangel“ im Gerät deaktiviert war.

**Ursache:** Im Gerät war ein interner Bewässerungsplan aktiv, der unabhängig von der Home-Assistant-Steuerung lief. Nachweisbar wurde das über die Rohdaten des Geräts in Zigbee2MQTT, die einen laufenden Zeitplan mit erwartetem Endzeitpunkt auswiesen.

**Lösung – zweistufig:**

- Ursache beseitigen: Der zyklische Bewässerungsplan wird per MQTT-Nachricht direkt am Gerät auf null zurückgesetzt.
- Absicherung: Eine Watchdog-Automation prüft alle 9 Minuten, ob das Ventil innerhalb der Betriebszeit unerwartet ausgeschaltet ist, und schaltet es gegebenenfalls wieder ein. Die maximale Unterbrechung ist damit auf 9 Minuten begrenzt.

**Bewertung:** Der Watchdog behebt nicht die Ursache, sondern begrenzt die Auswirkung. Diese Redundanz wurde bewusst beibehalten, weil das Verhalten in der Geräte-Firmware liegt und bei einem Firmware-Update erneut auftreten kann.

## 3. Schreiblast-Analyse der Datenbank

**Ausgangsfrage:** Welche Komponenten erzeugen die Schreibzugriffe auf die Home-Assistant-VM, und wie stark ist der zugrundeliegende NVMe-Speicher dadurch ausgelastet?

**Vorgehen:** Die Analyse erfolgte zweistufig – auf Hypervisor-Ebene über die Blockstatistik der virtuellen Disk, in Home Assistant über eine direkte Abfrage der Recorder-Datenbank, die Zustandsänderungen je Entität in einem definierten Zeitfenster zählt.

**Befund:** Rund 38.750 Zustandsänderungen pro Stunde ( $\approx 10,8/\text{Sekunde}$ ). Die Verteilung war stark ungleich; die beiden größten Blöcke stammten aus der Wechselrichter- und der Batterieanbindung. Entscheidend war eine strukturelle Erkenntnis: Das Wechselrichter-Gateway wurde zwar korrekt alle 10 Sekunden abgefragt, stellte dabei aber 33 Einzelentitäten bereit – ein einzelner Abfragezyklus erzeugte damit 33 Datenbankschreibvorgänge.

**Maßnahme:** Die 24 Detailentitäten der einzelnen Wechselrichter-Kanäle wurden deaktiviert, die 9 Kernwerte



des Gateways blieben aktiv.

Ergebnis (gemessen, nicht geschätzt):

|                      | vorher  | nachher | Veränderung |
|----------------------|---------|---------|-------------|
| Wechselrichter-Block | 8.998/h | 2.478/h | –73 %       |

Bewertung der Gesamtsituation: Die parallel durchgeführte Live-Messung der Disk-Last ergab rund 26 Schreib-IOPS bei 0,31 MB/s und einer mittleren Schreiblatenz von etwa 3,1 ms. Gemessen an der Leistungsfähigkeit des NVMe-Speichers entspricht das einer Auslastung von deutlich unter 1 Prozent. Trotz zehntausender Zustandsänderungen pro Stunde ist der Speicher damit weit von einer Belastungsgrenze entfernt – die Schreibvorgänge werden vom Recorder effizient gebündelt.

Konsequenz: Weitere Optimierungen (etwa an den Batterie-Detailwerten als zweitgrößtem Block) sind technisch möglich, aber nicht erforderlich. Die Maßnahme wurde umgesetzt, weil sie sinnvoll und ohne Funktionsverlust möglich war – nicht, weil ein Engpass bestand. Für eine belastbare Aussage über den Gesamteffekt steht eine erneute Langzeitmessung aus.

## Systemgrenzen

Bewusst nicht Gegenstand dieses Dokuments:

| Bereich                                                        | Wo dokumentiert                                       |
|----------------------------------------------------------------|-------------------------------------------------------|
| Batteriespeicher, Omnibattery-Steuerung, Speicher-Automationen | eigene Dokumentation (30 Energie/01 Marstek Speicher) |
| Photovoltaik, Wechselrichter, Ertragsprognose                  | 30 Energie/02 Photovoltaik                            |
| Strombezug, Tarif, Verbrauchsauswertung                        | 30 Energie/04 Strombezug                              |
| Hypervisor, Server-Hardware, Netzwerk                          | 10 Hardware                                           |

Home Assistant ist für diese Bereiche die gemeinsame Steuer- und Datenebene; die fachliche Tiefe liegt jeweils in der genannten Dokumentation.

## Stand und Weiterentwicklung

Stand: Das System ist produktiv. Die Zigbee-Infrastruktur ist auf Zigbee2MQTT umgestellt und aus der HA-VM herausgelöst; die beschriebenen Problemstellungen sind gelöst bzw. wirksam begrenzt.

Offene Punkte:

- Erneute Langzeitmessung der Gesamt-Schreiblast zur Bestätigung des Optimierungseffekts.
- Optionale Bereinigung der zurückliegenden, durch den Faktor-100-Fehler verfälschten Kostenwerte über die Statistik-Korrektur.