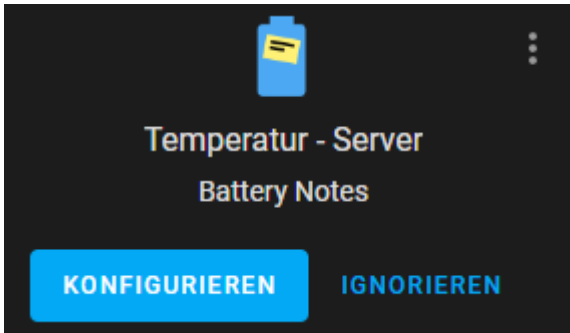


# Integrationen

- [Battery Notes +](#)

# Battery Notes +



## Automationen:

### Batterie wird leer:

```
alias: Battery Low Notification
description: Battery Low Notification with auto dismiss
mode: queued
triggers:
- event_type: battery_notes_battery_threshold
  event_data:
    battery_low: true
  id: low
  alias: Battery went low
  trigger: event
- event_type: battery_notes_battery_threshold
  event_data:
    battery_low: false
  id: high
  alias: Battery went high
  trigger: event
conditions: []
actions:
- choose:
  - conditions:
    - condition: trigger
```

```

    id:
      - low
sequence:
  - data:
      title: |
        {{ trigger.event.data.device_name }} Battery Low
      notification_id: >-
        {{ trigger.event.data.device_id }}-{{
          trigger.event.data.source_entity_id }}
      message: >
        Das Gerät hat {{ trigger.event.data.battery_level }}%
        verbleibend. {{ '\n' -}} Du brauchst {{
          trigger.event.data.battery_quantity }}× {{
          trigger.event.data.battery_type }}
      action: persistent_notification.create
  - conditions:
      - condition: trigger
        id:
          - high
sequence:
  - data:
      notification_id: >-
        {{ trigger.event.data.device_id }}-{{
          trigger.event.data.source_entity_id }}
      action: persistent_notification.dismiss

```

## Batterie kein Statusupdate:

[Blueprint](#)

## Batterie ersetzt:

[Blueprint](#)

## Battery State Card:

vorab müsst ihr in HACS noch



## Battery State Card / Entity Row

Battery state card for Home Assistant

installieren und dann könnt ihr folgenden Code als Card einfügen:

```
type: custom:battery-state-card
secondary_info: '{attributes.battery_type_and_quantity}'
round: 0
filter:
  include:
    - name: entity_id
      value: '*_battery_plus'
  exclude:
    - name: attributes.battery_low
      value: false
bulk_rename:
  - from: "Battery+"
sort:
  - state
```