

WLFRED – MQTT-/JSON-Datenmodell

Zwischenstand 23.08.2026 – Rüdiger Bäcker

Zusammenführung der bisher beschlossenen JSON-Strukturen mit den Informationen des heutigen Gesamt-XML des WLFRED.

1. Grundsätze

- Es wird zwischen config (dauerhaft gespeicherte Einstellungen) und status (aktuelle Betriebs-/Zustandswerte) unterschieden.
- Lesen erfolgt einheitlich mit cmd = get. Schreiben erfolgt mit cmd = set.
- Schreiben von config ist nur bei aktivem Konfigurationsmodus erlaubt.
- Außerhalb des Konfigurationsmodus sind nur ausdrücklich als public_read freigegebene Datensätze lesbar.
- Funktionsdaten umfassen ausschließlich F0 bis F16.
- WLAN-Passwörter werden nicht im Klartext übertragen. Das konkrete Schutzverfahren für credential wird separat festgelegt.

2. Zugriffsklassen

Klasse	Bedeutung	Regel
public_read	Immer lesbar	Lesen im Normal- und Konfigurationsbetrieb
config_read	Geschützt	Lesen nur bei aktivem Konfigurationsmodus
config_write	Schreiben	Nur bei aktivem Konfigurationsmodus

3. Übersicht der Datensätze

Kategorie	item	Lesen	Inhalt
config	name	public_read	WLFRED-Name
config	vehicle_number	public_read	Triebfahrzeugnummer
config	server	config_read	WiThrottle-Server
config	wifi	config_read	gespeichertes WLAN
config	loco_address	public_read	DCC-Adresse
config	loco_address_mode	public_read	kurze/lange Adresse
config	speed_steps	config_read	Fahrstufenmodus
config	direction	config_read	Richtungskonfiguration
config	functions	config_read	F0 bis F16
config	poti_calibration	config_read	Poti-Kalibrierung
config	battery_calibration	config_read	Batterie-Kalibrierung
config	center_function	config_read	Center-Switch/-Function
config	owner_mode	config_read	Besitzermodus
config	wifi_roaming	config_read	Roaming ein/aus
config	roaming_config	config_read	Roaming-Parameter
status	device_status	public_read	IP, Firmware, MAC
status	battery_status	public_read	Akkuspannung und Warnzustand
status	wifi_status	public_read	aktuelle WLAN-Verbindung
protocol	protocol_info	public_read	Struktur-/Protokollversion

4. Einheitlicher Lese-/Schreibmechanismus

Beispiel für einen Leseaufruf:

```
{
  "cmd": "get",
  "data": {
    "item": "vehicle_number"
  }
}
```

Beispiel für einen Schreibaufruf (nur bei aktivem Konfigurationsmodus):

```
{
  "cmd": "set",
  "data": {
    "item": "vehicle_number",
    "value": {
      "vehicle_number": "V 36 316"
    }
  }
}
```

Ist ein Datensatz außerhalb des Konfigurationsmodus nicht lesbar, wird eine definierte Fehlermeldung zurückgegeben:

```
{
  "cmd": "error",
  "data": {
    "item": "wifi",
    "error": "config_mode_required"
  }
}
```

5. config – 15 Konfigurationsdatensätze

5.1 name

Zugriff: public_read

Name des WLFRED.

```
{
  "cmd": "name",
  "access": "public_read",
  "data": {
    "name": "Rbaec#V36315#e1611e"
  }
}
```

5.2 vehicle_number

Zugriff: public_read

Neue, frei lesbare Triebfahrzeugnummer; als String, z. B. V 36 316.

```
{
  "cmd": "vehicle_number",
  "access": "public_read",
  "data": {
    "vehicle_number": "V 36 316"
  }
}
```

5.3 server

Zugriff: config_read

WiThrottle-Servername bzw. Host, Port und automatische Serversuche.

```
{
  "cmd": "server",
  "access": "config_read",
  "data": {
    "host": "undef",
    "port": 12090,
    "auto_discovery": true
  }
}
```

5.4 wifi

Zugriff: config_read

Gespeichertes WLAN. credential ist geschützt und niemals Klartext. Bei mehreren WLANs wird network_id verwendet.

```
{
  "cmd": "wifi",
  "access": "config_read",
  "data": {
    "network_id": 1,
    "ssid": "xxpdpxx",
    "credential": "<geschützt>",
    "enabled": true
  }
}
```

5.5 loco_address

Zugriff: public_read

DCC-Adresse des zugeordneten Triebfahrzeugs.

```
{
  "cmd": "loco_address",
  "access": "public_read",
  "data": {
    "address": 36
  }
}
```

5.6 loco_address_mode

Zugriff: public_read

Kennzeichnet kurze bzw. lange DCC-Adresse.

```
{
  "cmd": "loco_address_mode",
  "access": "public_read",
  "data": {
    "long_address": false
  }
}
```

5.7 speed_steps

Zugriff: config_read

Fahrstufenmodus. Das heutige Gesamt-XML liefert hierfür Mode.

```
{
  "cmd": "speed_steps",
  "access": "config_read",
  "data": {
    "mode": ""
  }
}
```

5.8 direction

Zugriff: config_read

Richtungskonfiguration entsprechend dem heutigen Feld Direction.

```
{
  "cmd": "direction",
  "access": "config_read",
  "data": {
    "direction": 2
  }
}
```

5.9 functions

Zugriff: config_read

Funktionszuordnung ausschließlich F0 bis F16.

```
{
  "cmd": "functions",
  "access": "config_read",
  "data": {
    "F0": 0,
    "F1": 0,
    "F2": 0,
    "F3": 0,
    "F4": 0,
    "F5": 0,
    "F6": 0,
    "F7": 0,
    "F8": 0,
    "F9": 0,
    "F10": 0,
    "F11": 0,
    "F12": 0,
    "F13": 0,
    "F14": 0,
    "F15": 0,
    "F16": 0
  }
}
```

5.10 poti_calibration

Zugriff: config_read

Interne, dauerhaft gespeicherte Kalibrierung des Fahrregler-Potentiometers.

```
{
  "cmd": "poti_calibration",
  "access": "config_read",
  "data": {
    "min": 120,
    "max": 3950
  }
}
```

5.11 battery_calibration

Zugriff: config_read

Interner Kalibrierfaktor der Batteriespannungsmessung.

```
{
  "cmd": "battery_calibration",
  "access": "config_read",
  "data": {
    "factor": 1.0
  }
}
```

5.12 center_function

Zugriff: config_read

Abbildung des heutigen centerSwitch-/Center-Function-Parameters.

```
{
  "cmd": "center_function",
  "access": "config_read",
  "data": {
    "value": -2
  }
}
```

5.13 owner_mode

Zugriff: config_read

Besitzermodus ein/aus. Außerhalb des Konfigurationsmodus nicht lesbar.

```
{
  "cmd": "owner_mode",
  "access": "config_read",
  "data": {
    "enabled": true
  }
}
```

5.14 wifi_roaming

Zugriff: config_read

WLAN-Roaming ein/aus. Außerhalb des Konfigurationsmodus nicht lesbar.

```
{
  "cmd": "wifi_roaming",
  "access": "config_read",
  "data": {
    "enabled": true
  }
}
```

5.15 roaming_config

Zugriff: config_read

Künftige, getrennte Roaming-Parameter, soweit sie konfigurierbar werden.

```
{
  "cmd": "roaming_config",
  "access": "config_read",
  "data": {
    "min_rssi_difference": 10,
    "stability_time_s": 15,
    "scan_interval_s": 10
  }
}
```

6. status – aktuelle Betriebswerte

6.1 device_status

Zugriff: public_read

Aktuelle Geräteinformationen aus dem heutigen Gesamt-XML. Diese Werte sind keine dauerhaft zu ändernde Konfiguration.

```
{
  "cmd": "device_status",
  "access": "public_read",
  "data": {
    "local_ip": "192.168.1.99",
    "firmware_revision": "WLFRED 1C 20260822 12 BM-KS-RS-MQTT-Q1 RB",
    "mac_address": "ecda3be1611e"
  }
}
```

6.2 battery_status

Zugriff: public_read

Aktueller Akkustand. batteryLow aus dem heutigen Gesamt-XML wird als boolescher Wert ausgegeben.

```
{
  "cmd": "battery_status",
  "access": "public_read",
  "data": {
    "voltage_mv": 3559,
    "low": false
  }
}
```

6.3 wifi_status

Zugriff: public_read

Aktueller WLAN-Zustand. SSID hier beschreibt das tatsächlich verbundene WLAN, nicht die gespeicherte WLAN-Konfiguration.

```
{
  "cmd": "wifi_status",
  "access": "public_read",
  "data": {
    "connected": true,
    "ssid": "xxpdpxx",
    "signal_strength_dbm": -58
  }
}
```

7. Protokoll-/Strukturinformation

7.1 protocol_info

Zugriff: public_read

Übernimmt die Funktion des heutigen strukturVersion-Feldes und ermöglicht Versionsprüfung zwischen WLFRED und Gegenstelle.

```
{
  "cmd": "protocol_info",
  "access": "public_read",
  "data": {
    "structure_version": 1
  }
}
```

8. available – aktuell lesbare Datensätze

Eine Gegenstelle kann abfragen, welche Datensätze im aktuellen Betriebszustand verfügbar sind. Im Normalbetrieb ergibt sich derzeit folgende Liste:

```
{
  "cmd": "available",
  "data": {
    "config_mode": false,
    "config_items": [
      "name",
      "vehicle_number",
      "loco_address",
      "loco_address_mode"
    ],
    "status_items": [
      "device_status",
      "battery_status",
      "wifi_status"
    ],
    "protocol_items": [
      "protocol_info"
    ]
  }
}
```

Im aktiven Konfigurationsmodus werden zusätzlich alle config_read-Datensätze als lesbar gemeldet. Änderungen an config bleiben ausschließlich in diesem Modus zulässig.

9. Abbildung des heutigen Gesamt-XML

Heutiges XML	Künftiges JSON
strukturVersion	protocol_info.structure_version
throttleName	name.name
localIP	device_status.local_ip
firmwareRevision	device_status.firmware_revision
batteryVoltage	battery_status.voltage_mv
batteryLow	battery_status.low
WiFi/Connected	wifi_status.connected
WiFi/SSID	wifi_status.ssid
WiFi/signalStrength	wifi_status.signal_strength_dbm
WiFi/macAddress	device_status.mac_address
DCCadress	loco_address.address
Mode	speed_steps.mode
Direction	direction.direction
LongAdress	loco_address_mode.long_address
FUNCTIONS 0-16	functions.F0 ... functions.F16
NETWORKS/SSID	wifi.ssid
NETWORKS/Enabled	wifi.enabled
LOCOSERVER/ServerName	server.host
LOCOSERVER/Port	server.port
LOCOSERVER/Automatic	server.auto_discovery
centerSwitch	center_function.value

10. Noch offene Festlegungen

- Konkretes kryptografisches Verfahren für die geschützte WLAN-Credential-Übertragung.
- Endgültige MQTT-Topic-Struktur.
- Endgültiges Antwort-/Fehlerformat einschließlich Request-ID, falls erforderlich.
- Genaue Semantik einzelner Altparameter wie Mode, Direction und centerSwitch, bevor die Implementierung festgeschrieben wird.
- Ob weitere Statuswerte, z. B. aktuelle BSSID, WiThrottle-Verbindungsstatus oder Fahrzustand, zusätzlich aufgenommen werden.

Entwurf RBaEC