

Unified-E WebHttp-JSON Adapter Benutzerhandbuch

Web-Endpunkte und Datenpunkte konfigurieren

Software-Version 3.1.0.0, zuletzt aktualisiert: Juli 2025

Herausgeber: Unified-E AG, Winterthur, Schweiz



Inhalt

1	Allgemeines	3
2	Allgemeine Parameter konfigurieren	3
3	Webserver Schnittstelle für Datenpunkte.....	4
3.1	Web-Schnittstelle «Standard»	4
3.1.1	Zugriffstyp «JSON-Objekt lesen»	4
3.1.2	Zugriffstyp «Datenpunkt-URL für «lesen/schreiben»	5
3.2	Web-Schnittstelle «Unified-E (vordefiniert)»	6
3.2.1	Spezifische Parameter	6
3.2.2	Erforderliche REST-Aktionen	7

1 Allgemeines

Mit Hilfe des WebHttp-JSON Adapters kann auf Datenpunkte zugegriffen werden, die von einem Webserver verwaltet werden. Damit können beispielsweise Daten von bereits etablierten Geräten ausgelesen werden oder auch eine Brücke zum ERP-System realisiert werden.

Es werden zwei grundsätzliche Web-Schnittstellen angeboten:

- **Standard:**
Der Zugriff erfolgt über Standard-Aufrufe, welcher bereits von vielen Geräten unterstützt wird, die Daten über einen Webserver bereitstellen. Eine zusätzliche Webserver-Programmierung ist hier nicht erforderlich.
- **Unified-E (vordefiniert):**
Hier ist eine Programmierung der REST-Schnittstelle bei einem Webserver erforderlich. Es stehen erweiterte Funktionen im Vergleich zur «Standard»-Web-Schnittstelle zur Verfügung:
 - Authentifizierung über Benutzername und Passwort
 - Datenpunkt-Kontexte: Hier kann der Webserver abhängig vom Bediengerät-Namen/Benutzer oder abhängig von der HMI-App Sprache unterschiedliche Werte zurückliefern

Beide Web-Schnittstellen werden im Kapitel 3 ausführlich beschrieben.

2 Allgemeine Parameter konfigurieren

Folgende allgemeinen Eigenschaften können beim WebHttp-JSON Adapter gesetzt werden.

Adresse:

Die Adresse des Web-Server (Endpunktes), z. B. die IP-Adresse, URL oder localhost.

Parameter-Einstellungen:

- **Web-Schnittstelle:** Beschreibt die Art und Weise, wie der Endpunkt-Adapter aus dem Webserver Datenpunkte liest oder schreibt:
 - **Standard:** Einfaches Standard-JSON Format – siehe Kapitel 3.1
 - **Unified-E (vordefiniert):** Von Unified-E definierte REST-Funktionen für erweiterte Lese/Schreib-Möglichkeiten – siehe Kapitel 3.2
- **Protokoll:** Web-Protokoll, HTTP oder HTTPS
- **Timeout (ms):** Der Timeout-Wert für den Aufruf der REST-Schnittstelle in Millisekunden

Weitere spezifische Parameter, abhängig von der ausgewählten Web-Schnittstelle, sind weiter unten beschrieben.

3 Webserver Schnittstelle für Datenpunkte

Grundsätzlich erfolgt der Zugriff auf Variablen symbolisch. Wird im Unified-E App Designer in der Datenpunkte-Tabelle keine Adresse beim Datenpunkt gesetzt, dann wird für die Adressierung die Datenpunkt-Bezeichnung (Spalte «Bezeichnung») verwendet.

Abhängig vom Adapter-Parameter «Web-Schnittstelle» holt der Endpunkt-Adapter die Datenpunkt-Werte auf unterschiedliche Weise vom Web-Server. Die unterschiedlichen REST-Schnittstellen sind in den folgenden Unterkapiteln genau beschrieben.

3.1 Web-Schnittstelle «Standard»

Diese JSON-Schnittstelle wird bereits von vielen Geräten unterstützt. Der Endpunkt-Adapter kann daher direkt vom vorhandenen Web-Server des Geräts (Endpunkt) die gewünschten Datenpunkt-Werte auslesen. Es werden zwei Zugriffstypen unterstützt, um eine möglichst breite Palette von Geräten mit Webserver zu unterstützen. Der Aufbau der Datenpunkt-Adresse ist abhängig vom Zugriffstyp.

3.1.1 Zugriffstyp «JSON-Objekt lesen»

Bei diesem Zugriffstyp können Einzelwerte aus einem umfangreichen JSON-Objekt gelesen werden – ein Schreiben ist nicht vorgesehen.

Ein GET-Aufruf liefert ein gesamtes JSON-Objekt (z. B. viele Werte eines Geräts). Die gewünschten Werte des JSON-Objekts können Datenpunkten zugeordnet werden.

Der Parameter 'Lesen URL-Pfaderweiterung' ist optional und ermöglicht eine URL-Erweiterung zum JSON Objekt. Um auf die enthaltenen Werte zuzugreifen, muss in der Datenpunkte-Tabelle unter der Spalte «Adresse» der Name bzw. Pfad zum JSON-Wert eingetragen werden.

URL zum JSON-Objekt im Web-Server:

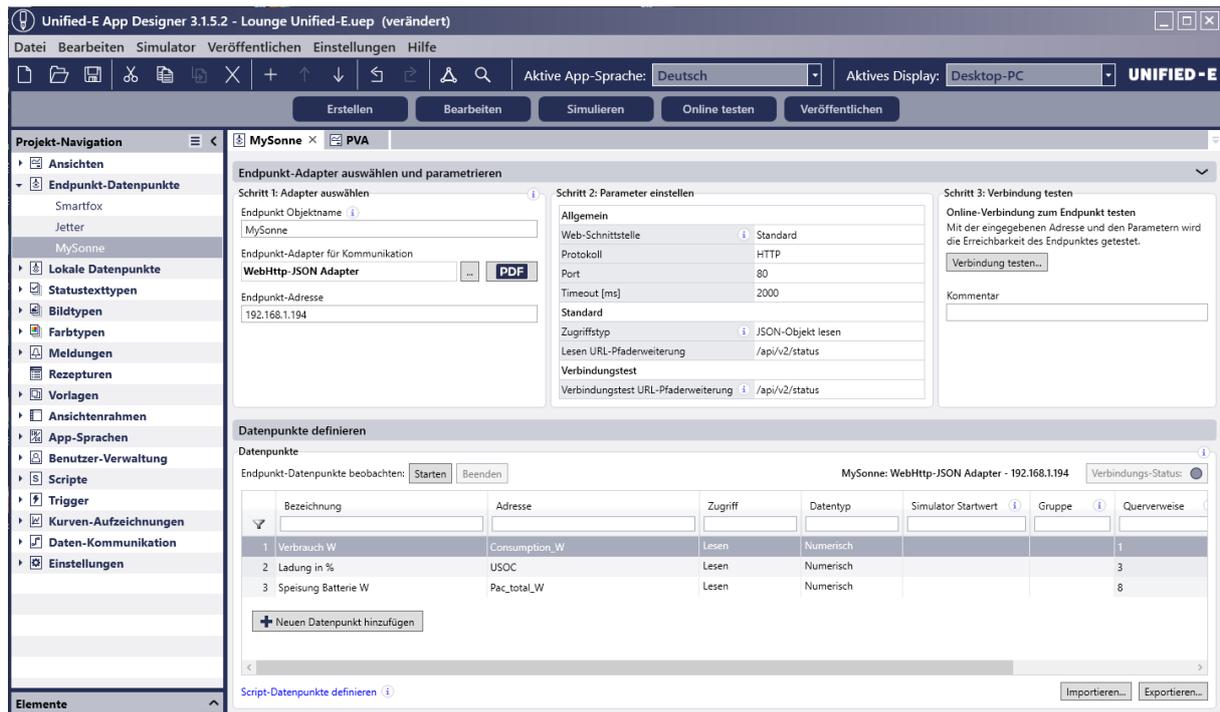
Die URL zum JSON-Objekt setzt sich aus der Endpunkt-Adresse und den Endpunkt-Parametern «Protokoll» und «Lesen URL-Pfaderweiterung» wie folgt zusammen:

<Protokoll>://<Endpunkt-Adresse>:<Port>/<Lesen URL-Pfaderweiterung>

Datenpunkt-Adresse (Wert in «Adresse»-Spalte):

Hier ist der Pfad des JSON-Wertes oder der Name einzutragen, z. B. «Temperature» oder «TemperatureSensor[0].Value»

Beispiel:



3.1.2 Zugriffstyp «Datenpunkt-URL für «lesen/schreiben»

Für jeden Datenpunkt wird beim Lesen ein GET-Aufruf und beim Schreiben ein PUT- bzw. POST-Aufruf ausgeführt. Die URL-Pfaderweiterung zum Datenpunkt wird direkt in der Tabelle Datenpunkte in der Spalte Adresse eingetragen.

URL zum Datenpunkt:

Die URL wird wie folgt zusammengesetzt:

<Protokoll>://<Endpunkt- Adresse>:<Port>/<Datenpunkt URL-Pfaderweiterung (aus Spalte 'Adresse' in der Tabelle)>

URL-Beispiel:

- Endpunkt-Adresse: 192.168.1.20
- Endpunkt-Parameter
 - Protokoll: HTTP
 - Port: 80
 - Datenpunkt URL-Pfaderweiterung: DataPoints
 - Adresse in Datenpunkte-Tabelle: «Sensor1/Temperature»

Der Datenpunkt wird dann unter folgenden URL vom Adapter gelesen oder geschrieben:

http://192.168.1.20:80/DataPoints/Sensor1/Temperature

Unterstützte Lesen-Formate:

Es wird ein einfaches JSON-Objekt der Form {<Name>: <Wert>} oder ein einfacher Text/String mit dem Wert erwartet.

Beispiele:

- {"Temperatur": 30}
- 30

JSON-Format beim Schreiben:

{"<Datenpunkt URL-Pfaderweiterung>": <Wert>}

Beispiel: {"Temperatur": 30}

Parameter «Lesen/Schreiben Anfrage-Aktion»:

Mit diesem Parameter wird festgelegt, ob das Schreiben mit der PUT oder POST-Aktion erfolgen soll.

3.2 Web-Schnittstelle «Unified-E (vordefiniert)»

Bei dieser Web-Schnittstelle werden vom Webserver folgende REST-Aktionen erwartet:

- GET-Aktion:
 - `IsAlive`: Prüfung der Erreichbarkeit des Endpunktes
- PUT-Aktionen für den Datenpunkte-Zugriff:
 - `ReadValues`: Datenpunkte auslesen
 - `WriteValues`: Datenpunkte schreiben
 - `ReadDataTables`: Datenpunkt-Tabelle auslesen

Diese REST-Aktionen sind speziell für die Kommunikation mit Unified-E optimiert unten im Kapitel 3.2.2 genauer beschrieben.

3.2.1 Spezifische Parameter

Folgende spezifischen Parameter stehen beim Endpunkt-Adapter zu Verfügung:

- **Namensbereich**: Auch als Ressourcenpfad bekannt. Beispiel-URL:
`http://domain.com:80/Namensbereich/Aktion`
- **Authentifizierung**: Bestimmt ob bzw. Authentifizierungsart verwendet werden soll
 - **Keine Authentifizierung**: Der Web-Endpunkt erfordert keine Authentifizierung.
 - **Benutzerdefinierte Authentifizierung**: Die Authentifizierung erfolgt bei jedem REST-Aufruf. Benutzername und Passwort werden bei jedem Aufruf mitgeführt

- 'Http Basic Auth'-Authentifizierung: Das Standard-Protokoll "Http Basic Auth" wird benutzt
- **Datenpunkt-Kontext:** Legt fest, ob der Datenpunkt global vorhanden ist oder pro Benutzer:
 - **Global:** Alle Benutzer erhalten den gleichen Wert, wenn sie einen bestimmten Datenpunkt anfordern. Dies ist typisch für Datenpunkte, die Maschinenzustände darstellen
 - **Benutzer:** Verschiedene Benutzer können unterschiedliche Werte erhalten, wenn sie einen bestimmten Datenpunkt anfordern. Der Endpunkt muss den Benutzername-Wert der JSON-Anfrage auswerten
 - **Sprache:** Alle Benutzer einer Sprache erhalten denselben Datenpunktwert für einen bestimmten Datenpunkt. Dieser Kontext ist sinnvoll, wenn Datenpunkte vom Typ 'Text' mehrsprachig sein sollen. Die REST-Aktion muss hierfür die LanguageId auswerten

3.2.2 Erforderliche REST-Aktionen

Allgemeine JSON-Eigenschaften:

Folgende JSON-Eigenschaften sind bei allen Request-Aktionen verfügbar:

- **EndPointUser:** Benutzername für die benutzerdefinierte Authentifizierung
- **EndPointPassword:** Passwort für die benutzerdefinierte Authentifizierung
- **User:** Der Name des App-Benutzers bzw. Name des registrierten Bediengeräts bei Gateway-Kommunikation. Dieser Wert kann vom Webserver genutzt werden, um benutzerspezifische Datenpunkte oder Werte bereitzustellen (siehe «Datenpunkt-Kontext» Parameter)
- **LanguageId:** Sprachcode gemäss ISO 639-1 im 2-Buchstaben-Format (z. B. en für Englisch, de für Deutsch). Der Webserver kann diesen Wert auswerten, um sprachabhängige Inhalte zurückzugeben (siehe «Datenpunkt-Kontext»-Parameter)

3.2.2.1 IsAlive-Aktion

Der GET-Aufruf wird ohne Parameter aufgerufen und wird für die Überprüfung der grundsätzlichen Erreichbarkeit eines Endpunktes verwendet. Als Rückgabewert wird "true" erwartet.

3.2.2.2 ReadValues-Aktion

Liest mehrere Datenpunkte in einem einzigen REST-Aufruf.

- HTTP-Methode: PUT
- Pfad: /ReadValues

Request (JSON):

```
{
  "EndPointUser": "admin",
  "EndPointPassword": "secret",
  "User": "john.doe",
  "LanguageId": "de",
  "DataPoints": [
    { "DataPointAddress": "Sensor1/Temperature" },
    { "DataPointAddress": "Sensor2/State" }
  ]
}
```

Response (JSON):

```
[
  {
    "Value": "22.5",
    "AccessState": 0
  },
  {
    "Value": "1",
    "AccessState": 0
  }
]
```

AccessState-Werte:

- 0 – Wert gültig
- 1 – Datenpunkt nicht gefunden
- 2 – Fehler

3.2.2.3 WriteValues Aktion

Schreibt neue Werte für mehrere Datenpunkte.

- HTTP-Methode: PUT
- Pfad: /WriteValues

Request (JSON):

```
{
  "EndPointUser": "admin",
```

```
"EndPointPassword": "secret",
"User": "john.doe",
"LanguageId": "en",
"DataPointValues": [
  {
    "DataPoint": { "DataPointAddress": "Sensor1/Setpoint" },
    "Value": "45"
  },
  {
    "DataPoint": { "DataPointAddress": "Motor1/Start" },
    "Value": "1"
  }
]
```

Response (JSON):

„True“, wenn erfolgreich, sonst „False“.

3.2.2.4 ReadDataTables-Aktion

Liest tabellarische Daten von einem oder mehreren Datenpunkten (z. B. Messwertverläufe).

- HTTP-Methode: PUT
- Pfad: /ReadDataTables

Request (JSON):

```
{
  "EndPointUser": "admin",
  "EndPointPassword": "secret",
  "User": "john.doe",
  "LanguageId": "en",
  "DataPoints": [
    {
      "DataPointAddress": "Logger1/Temperatures",
      "LastModificationId": "abc123"
    }
  ]
}
```

}

Response (JSON):

```
[
  {
    "DataTable": {
      "Rows": [
        { "Items": ["2024-01-01T10:00:00", "21.3"] },
        { "Items": ["2024-01-01T10:01:00", "21.4"] }
      ]
    },
    "ModificationType": 2,
    "AccessState": 0,
    "ModificationId": "def456"
  }
]
```

ModificationType-Werte:

- 0 – Alle Zeilen wurden zurückgegeben (Reset)
- 1 – Keine Änderungen seit letzter Abfrage
- 2 – Neue Zeilen oben hinzugefügt
- 3 – Neue Zeilen unten hinzugefügt

AccessState-Werte:

AccessState-Werte siehe oben.