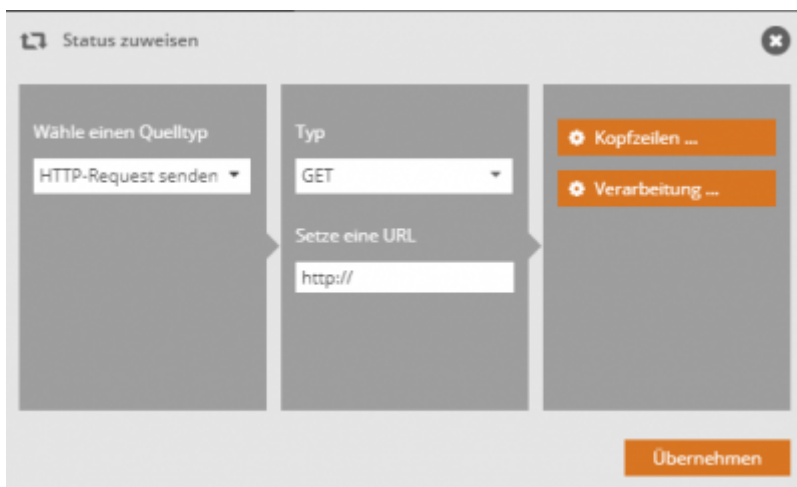


Status: HTTP-Request senden



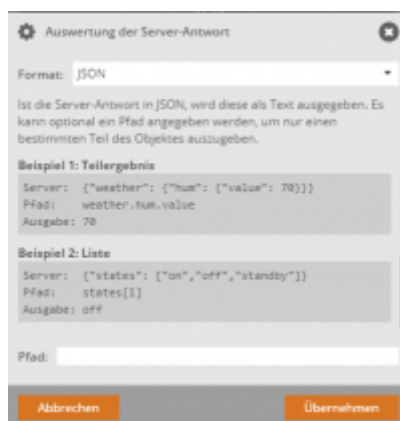
(Ab AIO CREATOR NEO-Version 2.8.5 oder höher.)

Wird als Status für z. B. ein Textelement *HTTP-Request senden* gewählt, verhält sich dies zunächst wie die gleichnamige Aktion (siehe [Aktion: HTTP-Request senden](#)). Jedoch wird die Server-Antwort als Statuswert an das Element weitergegeben. Das Textelement z. B. gibt dies dann als Text aus. Ein HTTP-Request für einen Status kann mit folgenden Elementen realisiert werden: Textelement, Button, Grafik, Slider/Rundslider und Analogmeter.

Wichtig Bitte beachten Sie, dass der Status eines HTTP-Requests mit Grafiken/Statusbuttons bzw. Slidern/Analogmetern nur dargestellt werden kann, wenn der HTTP-Request dem Wertebereich der jeweiligen Statusgrafik bzw. dem Wertebereich des Sliders/Analogmeters entspricht. Der Wertebereich von Statusgrafiken kann mit [Statusregeln](#) angepasst werden, der Wertebereich von Slidern/Analogmetern über Erweiterte Optionen → Intervall.

Wichtig: Bei jeder Statusaktualisierung wird auch erneut der HTTP-Request an den Server geschickt. Bitte bedenken Sie dies, wenn Sie z. B. eine Web API benutzen, die nur eine beschränkte Anzahl an Anfragen pro Stunde erlaubt.

Auswertung von JSON



Gibt der Server als Antwort ein JSON-Objekt zurück, wie es z. B. bei vielen Web APIs der Fall ist, kann dies speziell ausgewertet werden. Klicken Sie dazu im Dialog für *Status zuweisen* auf den Button

Verarbeitung. Daraufhin öffnet sich ein neues Fenster. Wählen Sie hier als Format *JSON* aus.

In dem Eingabefeld für *Pfad* können Sie dann angeben, welchen Teil des JSON-Objektes Sie als Status verwenden möchten. Dazu müssen Sie bereits wissen, wie eine typische Server-Antwort aussehen wird.

Beispiel 1: Teilergebnis

```
{
  "wetter": {
    "luftfeuchte": { "wert": 70 },
    "temperatur": { "wert": 24.4, "einheit": "°C" }
  }
}
```

Wenn die Server-Antwort wie oben aussieht, lässt sich der Temperatur-Wert herausfiltern, indem man als Pfad angibt:

```
wetter.temperatur.wert
```

Das Textelement gibt in dem Fall 24.4 aus.

Beispiel 2: Listen

```
{
  "messungen": [
    { "name": "aktuell", "temp": 24 },
    { "name": "gestern", "temp": 21 },
    { "name": "vorgestern", "temp": 23 }
  ]
}
```

Möchten Sie auf ein bestimmtes Element innerhalb einer Liste zugreifen, dann funktioniert dies über eckige Klammern und dem Index (beginnend bei 0 für das erste Element):

```
messungen[1].temp
```

Ausgabewert ist in dem Fall 21, da auf das 2. Element in der Liste zugegriffen wird.

Beispiel 3: Punkt im Namen

Der Punkt wird als Trennzeichen für die JSON-Struktur verwendet. Jedoch kann es sein, dass ein Name/Schlüssel ebenfalls einen Punkt beinhaltet. In diesem Fall können Anführungszeichen verwendet werden.

```
{
  "v1": { "name": "Version 1" },
  "v1.1": { "name": "Version 1.1" },
  "v2": { "name": "Version 2" }
}
```

Um an das Objekt von „v1.1“ zu gelangen, müssen Sie als Pfad nun eintragen:

```
"v1.1".name
```

Dies wird als Version 1.1 ausgegeben.

Hinweis: Anzahl tatsächlich ausgeschickter Anfragen

Befinden sich mehrere Elemente auf einer Seite, die genau denselben Aufruf ausschicken (d. h. sowohl die URL, Methode, als auch ggf. Anfrageinhalt und Kopfzeilen sind alle identisch), dann wird dieser Aufruf nur ein einziges Mal ausgeschickt und nicht für jedes Element einzeln. Die Server-Antwort wird dann an alle Elemente weitergegeben.

Die Art der Auswertung kann sich dabei jedoch unterscheiden. Ist die Server-Antwort ein JSON-Objekt, können so verschiedene Elemente über die Pfad-Angabe verschiedene Werte für sich heraussuchen – ohne dass dadurch zusätzliche Server-Anfragen anfallen.

From:

<http://doku.mediola.com/> - **Dokumentationen**

Permanent link:

<http://doku.mediola.com/doku.php?id=de:creator:ui-remote-status-http>

Last update: **2020/06/29 15:35**

