Amtiri

REFERENCE MANUAL

WebService Out

Generated on November 25, 2026

WebService Out calls a web address when a variable changes: a GET that passes values in the address, or a POST that sends them in a body. It suits systems that take data over plain HTTP, such as a reporting service, a notification gateway or another company's collection endpoint.

Module	What it does	Typical use
WebService Out (`WEB_SERVICE_OUT`)	Sends one HTTP request per change of a trigger variable, when an enabler allows it	Reporting a reading or an event to an outside system

The module also has the settings every module shares: whether it is enabled, its name, its status and message variables and their thresholds. They are described in Settings every module shares.

WebService Out

One module makes one kind of call. Each change of its trigger variable runs the enabler, builds the address (and, for a POST, the body) from Amtiri Script expressions, and sends the request, all on the thread that reported the change.

Settings

Key	Setting	Type	Default	Meaning
`variable`	Variable	variable code	none	The trigger: each change of its value sends one request. Its value is not sent unless the expressions below read it
`enabler`	Enabler	expression	empty	An expression that must give `true` for the request to be sent. Empty means always
`url`	URL	expression	none	An expression that gives the address, as text. Required: without it every request fails
`mode`	Mode	`GET` or `POST`	`GET`	The HTTP method
`body`	Body	expression	empty	For `POST`, an expression that gives the request body, as text
`timeoutMs`	Timeout [ms]	int	`10000`	How long a whole call, connecting included, may take. At least `1`: with `0` or less the module does not open

Addresses and bodies are expressions

`url` and `body` are Amtiri Script expressions, not text, so a fixed address is written between double quotes, and values are joined to it with `+`:

```
URL    "https://collect.example.com/api/v1/reading?tag=LIT_301&value=" + LIT_301
```

- **Numbers** are written the way the engine writes them: `2.5`, `12`, and very large or very small values in exponent form, such as `1.0E7`.
- **Nothing is encoded for you.** A value is joined as it is, and a character an address cannot hold, such as a space, makes the request fail. Send text that may contain spaces in a POST body instead.
- **Quotes inside text** are written `\"`, which is how a JSON body is built: `"{\"tag\":\"LIT_301\",\"value\":" + LIT_301 + "}"`.
- **An expression that fails**, for example because it reads a variable that holds no value, fails the request.

Behaviour

- **One request per change.** Every change of the trigger variable, when the enabler allows it, sends one request.
- **What is sent.** The module adds no header of its own: no `Content-Type` and no `Authorization`. The request carries only what Java's HTTP client adds by itself: `Host`, `User-Agent: Java-http-client/<version>` and, for a POST, `Content-Length`. An endpoint that insists on a content type or on an authorization header cannot be called with this module.
- **HTTP/2 first.** Over `https`, the client uses HTTP/2 when the server offers it, and HTTP/1.1 otherwise. Over plain `http`, it sends HTTP/1.1 with a request to upgrade (`Upgrade: h2c`), which a server is free to ignore.
- **The answer.** A status from 400 up is a failure. Anything below counts as delivered, redirects included: a redirect is not followed. The answer's body is not read into any variable.
- **Timeouts.** A call that has not been answered `timeoutMs` after it started, connecting included, gives up, and the request fails.
- **Failures are counted**, and the next request that goes through clears them.

> **Warning**
>
> The request is sent on the thread that reported the change of the trigger variable, which formulas and the other modules share. While the endpoint takes its time, they wait, up to `timeoutMs` on every change. Keep `timeoutMs` as short as the endpoint allows, and trigger on a variable that changes only as often as the endpoint needs.

> **Caution**
>
> An address or body that carries a token or a password is stored in the module record, which is copied, exported and transferred with the configuration. Prefer an endpoint that trusts the network the plant calls from.

Messages

A counted failure is a `WARNING` that reads `… (1 of 3)`, `… (2 of 3)` and becomes an `ERROR` reading `… after 3 attempts`, with the thresholds set in Settings every module shares. The engine's own messages are in Module messages.

When	Level	Text
Opening: the mode is neither `GET` nor `POST`	`ERROR`	`Configuration invalid: unsupported mode '{mode}'`
Opening: `timeoutMs` is `0` or less	`ERROR`	`Configuration invalid: timeout must be at least 1 ms, not {timeout}`
Opening: the enabler, URL or body does not compile	`ERROR`	`Configuration invalid: {reason}`
Sending: the server answers with status 400 or above	`WARNING`, then `ERROR`	`HTTP {status} from {url}`
Sending: the request could not be made or ran out of time	`WARNING`, then `ERROR`	`Request to {url} failed: {reason}`

- `{url}` is the address the URL expression gave, without the user name and password an address can carry before its host, and with the value after `password=`, `token=`, `secret=` or `apikey=` shown as `***`. When the expression itself fails, it is the expression as written, masked the same way.

Example

A GET that reports the level of tank 301 each time it changes, but only while the plant is running:

```
Variable   LIT_301
Enabler    PLANT_RUNNING
URL        "https://collect.example.com/api/v1/reading?tag=LIT_301&value=" + LIT_301
Mode       GET
```

A POST that sends the day's volume once a day, when the formula variable `FQ_501_DAY` takes its new value:

```
Variable      FQ_501_DAY
Enabler       (empty)
URL           "https://collect.example.com/api/v1/daily"
Mode          POST
Body          "{\"tag\":\"FQ_501\",\"volume_m3\":" + FQ_501_DAY + "}"
Timeout [ms]  5000
```

The server receives `{"tag":"FQ_501","volume_m3":1843.5}` with no content type. A server that answers 200 clears any earlier failure; one that answers 503 three times in a row leaves the module in `ERROR` with `HTTP 503 from https://collect.example.com/api/v1/daily after 3 attempts`.

Next steps

- Settings every module shares
- Module messages
- Module catalogue

This page describes Data Orchester engine 6.12.0.