SMARTASSIST REMOTE

AEMP 2.0 API

ISO 15143-3

Machine telematics data integration guide for dealers, rental operations and equipment owners.

This document covers the AEMP 2.0 (ISO 15143-3) API. It describes what the API provides, which machine data it returns, how to request access and authenticate, and the full reference for every supported endpoint.

Document version: 1.6

Issued: November 2026

Audience: Dealers, rental operations, equipment owners and their integration partners

Contents

1. API Overview **3**
What the AEMP 2.0 API is 3
Adherence to the ISO standard 3
What the API does 3
Primary use cases 4
Intended audience 4

2. AEMP 2.0 API Capabilities **5**
Supported endpoints 5
Key concepts 6
Machine-Data Compatibility 8

3. Getting Started **12**
Step 1 — Request API access 12
Step 2 — Understand what you are issued 12
Step 3 — Authentication and authorization 13
Step 4 — Make your first call 14
Response and error codes 15
Integration best practices 15

4. API Reference **16**
Conventions 16
4.1 Fleet Snapshot 17
4.2 Equipment Snapshot 19
4.3 Time Series — Cumulative Operating Hours 20
4.4 Time Series — Cumulative Idle Hours 21
4.5 Time Series — Cumulative Fuel Used 22
4.6 Time Series — Fuel Used In The Preceding 24 Hours 23
4.7 Time Series — Fuel Remaining Ratio 24
4.8 Time Series — DEF Remaining 25
4.9 Time Series — Engine Condition 26
4.10 Time Series — Fault Codes 27
4.11 Time Series — Locations 29
4.12 Time Series — Average Daily Engine Load Factors 30
4.13 Time Series — Switch Status 31

Appendix A — Endpoint Quick Reference **32**

Appendix B — Glossary **33**

Appendix C — Support **34**

1. API Overview

What the AEMP 2.0 API is

The AEMP 2.0 API is a standardized telematics data interface for heavy equipment and fleet management. It provides a consistent, interoperable way to retrieve machine data, so that fleet owners, rental companies and contractors can pull telematics information directly into the systems they already use — ERP, fleet management, maintenance, rental or business intelligence platforms.

Because the interface follows a published international standard, an integration built against it is not specific to one manufacturer or one telematics provider. The same data structures, field names and request patterns apply across every compliant provider, which means a single integration can serve a mixed fleet.

Adherence to the ISO standard

The API implements **ISO 15143-3**, the international standard for construction machinery and equipment telematics data interchange. Conformance to the standard governs three things:

- **Data formatting** — field names, data types and units of measure are defined by the standard rather than by the provider.
- **Structure** — machine records, snapshots and time series collections follow a prescribed shape.
- **Transmission protocol** — endpoints are exposed over HTTPS as REST resources, with standard authentication and standard response codes.

The practical benefit is portability. An application written to consume ISO 15143-3 data from this API can consume the same data from any other conforming provider with little or no change to the parsing logic.

What the API does

The API answers two distinct kinds of questions through two families of endpoints.

Family	Question it answers	Endpoints
Snapshot	What is the current state of my machines? Where are they, what are the hour meters reading, how much fuel is left, are any faults active?	Fleet Snapshot (all machines) Equipment Snapshot (one machine)
Time Series	How did a single machine behave over a period of time? When did it run, where did it go, how did the hour meter and fuel figures progress, what faults occurred and when?	Eleven endpoints, each returning one data type for one machine over a date range

Both families return the same categories of machine data. The difference is depth versus breadth: snapshots provide the latest value for many machines at once, while time series provide the full history of one value for one machine.

Primary use cases

The following are the patterns most integrations fall into.

Use case	How the API supports it
Hour-meter and utilization reporting	Pull cumulative operating hours on a schedule to drive utilization dashboards, rental billing, internal cost allocation or customer-facing usage reports.
Rental billing and invoice substantiation	Use operating hours and engine condition history to document exactly when a machine ran during a rental period, and to resolve billing disputes with evidence.
Preventive maintenance scheduling	Drive service intervals in an existing maintenance system using live hour-meter readings rather than manual readings.
Fault and downtime monitoring	Retrieve diagnostic trouble codes to trigger service workflows, prioritize shop scheduling, or watch for recurring faults across a model line.
Fuel and cost analysis	Combine fuel used, fuel remaining, and load factor data to analyze consumption, identify unusual usage, and support fuel tax credit reporting.
Asset location and security	Track machine locations for logistics and jobsite management and audit remote engine lock activity.
Mixed-fleet consolidation	Because the interface is standardized, machine data from this API can be merged with data from other conforming providers into one view.

Intended audience

This documentation is written for two groups:

- **Business and operations stakeholders** — Yanmar dealers, rental managers, service managers and equipment owners evaluating what the API can provide and deciding whether an integration is worth building. Sections 1 and 2 are written for this audience.
- **Developers and integration partners** — the technical staff or third-party vendor who will build and maintain the integration. Sections 3 and 4 are written for this audience.

No prior familiarity with the ISO standard is assumed. Working knowledge of HTTPS requests, REST APIs, and either JSON or XML is assumed for Sections 3 and 4.

2. AEMP 2.0 API Capabilities

This section describes which capabilities are supported, the concepts an integrator needs to understand before building, and which machine data is realistically available.

Supported endpoints

Thirteen endpoints are supported. Two return snapshots; eleven return time series.

#	Endpoint	Scope	Returns
1	**Fleet Snapshot**	Entire fleet	Latest known values for every machine in scope
2	**Equipment Snapshot**	One machine	Latest known values for a single machine, by VIN
3	**Cumulative Operating Hours**	One machine	Engine hour-meter readings over time
4	**Cumulative Idle Hours**	One machine	Idle hour readings over time
5	**Cumulative Fuel Used**	One machine	Lifetime fuel used readings over time
6	**Fuel Used In The Preceding 24 Hours**	One machine	Per-period fuel consumption
7	**Fuel Remaining Ratio**	One machine	Fuel tank level as a percentage
8	**DEF Remaining**	One machine	DEF tank level as a percentage
9	**Engine Condition**	One machine	Engine running / not running state changes
10	**Fault Codes**	One machine	Diagnostic trouble codes with SPN, FMI and description
11	**Locations**	One machine	Latitude, longitude and altitude of each reported position
12	**Average Daily Engine Load Factors**	One machine	Average engine load factor per day
13	**Switch Status**	One machine	Digital switch on/off changes (see Engine Restart Lock below)

> **IMPORTANT:** All eleven time series endpoints operate on a **single machine at a time**. There is no fleet-wide time series call. An integration that requires history for many machines must iterate through machines.

Key concepts

Resources

The API exposes two resource levels. `fleet` is the collection of all machines visible to your subscription. `fleet/equipment/{equipmentIdentifier}` is a single machine within that collection. Every time series endpoint is a sub-resource of a single machine. The API path is not case sensitive.

The equipment identifier

Wherever an endpoint takes an `equipmentIdentifier`, that value is the **machine VIN** — the vehicle identification number of the machine itself.

> **COMMON MISTAKE** The equipment identifier is **not** the serial number of the GPS or telematics device installed on the machine. Passing a device serial number will not resolve to a machine.

Events and data points

The standard treats each reported reading as a discrete data point carrying its own UTC timestamp. Two consequences follow from this:

- Data points appear only when the machine reported them. A machine that was powered down, out of cellular coverage, or not equipped to measure a given value produces no data points for that period — the result is an empty collection, not a zero value.
- Cumulative values (operating hours, idle hours, fuel used) are **lifetime running totals**, not per-period amounts. To calculate consumption or usage between two moments, subtract the earlier reading from the later one. Two endpoints are exceptions and already report per-period values: *Fuel Used In The Preceding 24 Hours* and *Average Daily Engine Load Factors*.

Date ranges

Every time series endpoint takes a start and an end timestamp in the request path. Both are full date-and-time values expressed in UTC (Zulu time), for example `2026-07-01T00:00:00Z`.

> **LIMIT** The window between `startDateUtc` and `endDateUtc` must be **14 days or less**. This is a constraint of the ISO specification, not a platform restriction. Retrieving a longer history requires splitting the request into consecutive 14-day windows.

Pagination

Fleet Snapshot and all time series endpoints are paginated. Results are returned **100 per page**, and the page size is fixed — it cannot be changed by the caller.

The page number is a required part of the request path, not an optional query parameter. **Pass `1` to retrieve the first page;** omitting it produces an invalid request.

Every paginated response includes a `Links` collection describing how to navigate the result set. Each entry has a `Rel` property naming the relationship and an `Href` property containing the URL:

```
{
   "Links":
   [
      { "Rel": "self", "Href": "https://temeda.azure-api.net/aemp/v2/fleet/2" },
      { "Rel": "prev", "Href": "https://temeda.azure-api.net/aemp/v2/fleet/1" },
      { "Rel": "next", "Href": "https://temeda.azure-api.net/aemp/v2/fleet/3" },
      { "Rel": "last", "Href": "https://temeda.azure-api.net/aemp/v2/fleet/12" }
   ]
}
```

The recommended pattern is to follow the `next` link until no `next` relationship is returned, rather than calculating page counts. A single machine's location history over a 14-day window can easily span several pages.

Request limits

The API is protected by rate limits and a daily quota. Both are applied per subscription and are shared across all AEMP 2.0 endpoints:

Limit	Value
Rate limit	120 requests per minute
Daily quota	20,000 requests per day

Exceeding either limit returns HTTP `403 Forbidden` together with a `Retry-After` header. The value of that header is the recommended wait, in seconds, before retrying. A well-behaved client should read `Retry-After` and back off rather than retrying immediately.

> **PLANNING NOTE:** Because time series calls are per-machine, quota consumption grows with fleet size. A 400-machine fleet pulling four time series endpoints daily uses 1,600 requests per day before pagination is counted. Budget the quota against fleet size and polling frequency before designing the integration.

Caching

Responses are cached for **15 minutes**. Repeated identical requests inside that window return the cached response rather than newly retrieved data. Polling an endpoint more frequently than every 15 minutes therefore consumes quota without returning fresher data.

Response formats

Every endpoint supports both JSON and XML. The format is selected with the `Accept` request header — it is not part of the URL:

Desired format	`Accept` header value
JSON	`application/json`
XML	`application/xml`

Machine-Data Compatibility

Supporting an endpoint and receiving data through it are two different things. An endpoint is available for every machine, but it returns data only where the machine measures and reports that value. The table below sets out current availability.

Data point	Availability	Notes
Asset information (make, model, identifier)	**Available**	Returned for all machines.
Location (latitude, longitude, altitude)	**Available**	Returned for all machines.
Cumulative operating hours	**Available**	Returned for all machines.
Engine condition	**Available**	Returned for all machines.
Fault codes (DTCs)	**Available**	It is supported by machines with devices equipped to read engine diagnostics, and Yanmar-specific fault descriptions are provided where applicable.
Cumulative fuel used	**Model dependent**	Reported by machines equipped to read engine diagnostics.
Fuel used in preceding 24 hours	**Model dependent**	Reported by machines equipped to measure fuel consumption.
Fuel remaining ratio	**Model dependent**	Requires a fuel level sensor on the machine.
DEF remaining	**Model dependent**	Applies only to machines with an SCR after-treatment system and a DEF level sensor.
Average daily engine load factor	**Model dependent**	Requires the machine to report engine load.
Cumulative idle hours	**Limited**	Only some models currently report this value, so results are frequently empty.
Switch status	**Available (Yanmar)**	Switch 1 is mapped to Engine Restart Lock. See below.
Cumulative distance travelled	**Not currently provided**	Not reported by the current machine population; the endpoint is therefore not documented here.
Cumulative power take-off hours	**Not currently provided**	Not reported by the current machine population; the endpoint is therefore not documented here.
Peak daily speed	**Model dependent**	Applicable only to a select range of machine models for which peak daily speed is relevant.

Switch Status and Engine Restart Lock

The ISO standard defines a switch generically. A switch data point carries a switch number and an on/off state, and nothing that describes what the switch controls.

On Yanmar machines, **switch number 1 is mapped to Engine Restart Lock**. Each time the engine is remotely locked or unlocked, that action is reported through Switch Status as a state change on switch 1 — reported as locked when the lock is in effect and unlocked when it is released.

This makes Switch Status the endpoint to use for programmatically auditing remote engine lock activity. The same history is also available in the SmartAssist Remote web and mobile applications for users who prefer to review it there.

Switch numbers other than 1 are not currently mapped to any function.

Machine model compatibility

The matrix below maps supported endpoints to machine models. Because data availability depends on what each model measures and reports, this matrix is the authoritative reference for what a given machine will return.

Endpoint	Data Element	Attribute	Tracked Carrier		Excavator			Electrical Excavator	Wheeled Excavator	Wheel Loader	Electrical Wheel Loader
			C30R*	C50R*	SV17*, ViO17*, SV19*, SV20*, ViO23-6*, ViO25-6*, SV26-6*, ViO26-6*, ViO27*-6*, ViO33-6*, ViO38-6*, ViO55-6A, ViO57-6A, SV60-A, B7-6	ViO25-7*, SV26-7*, ViO27-7*, ViO33-7*, ViO38-7*, SV86-7*, SV100-7*	ViO45*, ViO50*, ViO55-6B, ViO57-6B, SV60-B, ViO80-1*, ViO82, SV100-2A, SV120	SV17e	B75W, B95W, B110W	V70S, V80, TL80, V100, V120	V8e
Fleet Snapshot	EquipmentHeader	UnitInstallDateTime	✓	✓	✓	✓	✓	✓	✓	✓	✓
		OEMName	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Model	✓	✓	✓	✓	✓	✓	✓	✓	✓
		EquipmentID	✓	✓	✓	✓	✓	✓	✓	✓	✓
		SerialNumber	✓	✓	✓	✓	✓	✓	✓	✓	✓
		PIN	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Location	Latitude	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Longitude	✓	✓	✓	✓	✓	✓	✓	✓	✓
	EngineStatus	EngineNumber	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Running	✓	✓	✓	✓	✓	✓	✓	✓	✓
	FuelRemaining	Percent	-	-	-	✓	-	-	-	-	-
		FuelTankCapacityUnits	✓	✓	✓	✓	✓	✓	✓	✓	✓
		FuelTankCapacity	✓	✓	✓	✓	✓	✓	✓	✓	✓
	FuelUsed	FuelUnits	✓	✓	-	-	✓	-	✓	✓	-
		FuelConsumed	✓	✓	-	-	✓	-	✓	✓	-
	FuelUsedLast24	FuelUnits	✓	✓	-	-	✓	-	✓	✓	-
		FuelConsumed	✓	✓	-	-	✓	-	✓	✓	-
	DEFRemaining	Percent	-	✓	-	-	-	-	-	-	-
		DEFTankCapacityUnits	✓	✓	-	-	-	-	-	-	-
		DEFTankCapacity	✓	✓	-	-	-	-	-	-	-
	CumulativeOperatingHours	Hour	✓	✓	✓	✓	✓	✓	✓	✓	✓
	CumulativeIdleHours	Hour	-	-	-	-	-	-	-	-	-

Endpoint	Data Element	Attribute	Tracked Carrier		Excavator			Electrical Excavator	Wheeled Excavator	Wheel Loader	Electrical Wheel Loader
			C30R*	C50R*	SV17*, ViO17*, SV19*, SV20*, ViO23-6*, ViO25-6*, SV26-6*, ViO26-6*, ViO27*-6*, ViO33-6*, ViO38-6*, ViO55-6A, ViO57-6A, SV60-A, B7-6	ViO25-7*, SV26-7*, ViO27-7*, ViO33-7*, ViO38-7*, SV86-7*, SV100-7*	ViO45*, ViO50*, ViO55-6B, ViO57-6B, SV60-B, ViO80-1*, ViO82, SV100-2A, SV120	SV17e	B75W, B95W, B110W	V70S, V80, TL80, V100, V120	V8e
	AverageLoadFactorLast24	Percent	✓	✓	✓	✓	✓	✓	✓	✓	✓
	MaximumSpeedLast24	SpeedUnits	-	-	-	-	-	-	-	-	✓
		Speed	-	-	-	-	-	-	-	-	✓
TimeSeries	Cumulative Operating Hours		✓	✓	✓	✓	✓	✓	✓	✓	✓
	Cumulative Idle Hours		-	-	-	-	-	-	-	-	-
	Cumulative Fuel Used		✓	✓	-	-	✓	-	✓	✓	-
	Fuel Used In The Preceding 24 Hours		✓	✓	-	-	✓	-	✓	✓	-
	Fuel Remaining Ratio		-	-	-	✓	-	-	-	-	-
	DEF Remaining		-	✓	-	-	-	-	-	-	-
	Engine Condition		✓	✓	✓	✓	✓	✓	✓	✓	✓
	Fault Codes		✓	✓	✓	✓	✓	✓	✓	✓	✓
	Locations		✓	✓	✓	✓	✓	✓	✓	✓	✓
	Average Daily Engine Load Factors		✓	✓	✓	✓	✓	✓	✓	✓	✓
	Switch Status		✓	✓	✓	✓	✓	✓	✓	✓	✓

3. Getting Started

This section takes an integrator from no account to a first successful API call.

Step 1 — Request API access

API access is granted on request and centrally controlled. All access requests, at every level — dealer, rental operation or end customer — must go through **Yanmar Customer Support**.

1. **Submit a request to Yanmar Customer Support.** Include the requesting company name, the dealer or account the request relates to, a technical contact for the integration, and a short description of what the integration is intended to do.
2. **Yanmar reviews and approves the request.** Yanmar takes the corresponding measures to have the subscription provisioned against the correct account, so that the API returns only the machines that account is entitled to see.
3. **Receive your credentials.** On approval you are issued the credentials described in Step 2, along with the API base URL for your subscription.

> **NOTE** Access requests for Yanmar Telematics API must be submitted directly to **Yanmar Customer Support**, not through third-party telematics service providers or platforms used to integrate the Yanmar machine data.

Step 2 — Understand what you are issued

Two credentials are involved in every request, and they do different jobs:

Credential	Header	Purpose
Subscription key	`Ocp-Apim-Subscription-Key`	Identifies your subscription to the API gateway and is what the rate limit and daily quota are counted against. Constant for the life of the subscription.
Access token	`Authorization`	Identifies and authorizes the account whose machines you are entitled to see. Obtained by exchanging your client ID and client secret and expires after one hour.

Both must be present on every data request. A request with a valid subscription key but no valid token — or vice versa — is rejected.

Step 3 — Authentication and authorization

The platform supports two authentication methods. Which one applies depends on how the account was created.

Method	Applies to	How it works
Modern Auth (OAuth 2.0)	Accounts created from March 2021 onward — this includes all new dealer and customer subscriptions	Exchange a client ID and client secret for a short-lived access token, then send that token as a bearer token on each request.
Classic Auth	Legacy accounts created before March 2021	A static token is issued at provisioning and used indefinitely, alongside the subscription key.

New integrations use **Modern Auth**. If you are unsure which method applies to your account, ask Yanmar Customer Support before you begin.

Obtaining an access token (Modern Auth)

Post your client ID and client secret to the token endpoint, with `grant_type` set to `client_credentials`. The request is form-encoded, not JSON.

```
HTTP POST
https://temeda.azure-api.net/oauth2/auth/token

Content-Type: application/x-www-form-urlencoded

client_id={client_id}
&client_secret={client_secret}
&grant_type=client_credentials
```

The parameters are as follows:

Parameter	Value
`client_id`	(Your API User ID, provided with your credentials. It is formatted like an email address.)
`client_secret`	(Your API secret, provided with your credentials. Treat it like a password.)
`grant_type`	`client_credentials`

A successful response is a standard OAuth 2.0 token payload. The value of `access_token` is what subsequent data requests use:

```
{
  "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9...",
  "token_type": "Bearer",
  "expires_in": 3600
}
```

> **TOKEN LIFETIME** Tokens are valid for **one hour** (3600 seconds). No refresh token is issued: when a token expires, request a new one with the same call. A production integration should cache the token and reuse it until shortly before expiry — rather than requesting a fresh token for every call, which wastes quota and adds latency.

Step 4 — Make your first call

With a subscription key and an access token in hand, retrieve the first page of your fleet snapshot. This is the recommended first call: it confirms authentication, confirms entitlement scope, and shows you which machines the subscription can see.

```
HTTP GET
https://temeda.azure-api.net/aemp/v2/fleet/1

Accept: application/json
Authorization: Bearer eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9...
Ocp-Apim-Subscription-Key: 57b5cadfc7ba4cEX1234567890
```

A `200 OK` response containing an `Equipment` collection confirms that the integration is working end-to-end. From there:

- Note the identifier of a machine in the response — that value is the `equipmentIdentifier` for every single-machine and time series call.
- Follow the `Links` collection to page through the rest of the fleet.
- Move on to the time series endpoints in Section 4 for history on individual machines.

End-to-end summary

The complete path from no account to first result:

Stage	Action	Result
1	Submit an access request to Yanmar Customer Support	Request logged and reviewed
2	Yanmar approves and coordinates provisioning	Subscription created against the correct account
3	Receive client ID, client secret, subscription key and base URL	Ready to authenticate
4	POST client ID and client secret to the token endpoint	`access_token` valid for one hour
5	GET `fleet/1` with the token and subscription key	First page of fleet data returned

Response and error codes

The API uses standard HTTP status codes.

Code	Meaning	What to do
`200 OK`	The request succeeded.	Process the payload. An empty collection is a valid success response — it means the machine reported no data for that request.
`400 Bad Request`	The request is malformed.	Check the path. The most common causes are a missing page number, a date range longer than 14 days, and timestamps that are not valid UTC values.
`401 Unauthorized`	Authentication failed.	The token is missing, malformed or expired. Request a new token. Confirm the `Authorization` header includes the `Bearer` prefix.
`403 Forbidden`	Quota or rate limit exceeded, or the subscription is not entitled to the requested resource.	Read the `Retry-After` header and wait that many seconds before retrying. If the header is absent, check that the machine belongs to your account.
`404 Not Found`	The requested machine could not be found.	Confirm the `equipmentIdentifier` is the machine VIN, not a GPS device serial number, and that the machine is in scope for your subscription.
`429 Too Many Requests`	Too many requests in a short period.	Back off and retry. Reduce polling frequency; remember responses are cached for 15 minutes, so faster polling returns no fresher data.
`500 Internal Server Error`	An unexpected server-side error.	Retry after a short delay. If the error persists, contact Yanmar Customer Support with the request URL and timestamp.

Integration best practices

- **Poll no more often than every 15 minutes.** Responses are cached for that period, so more frequent polling consumes quota without returning newer data.
- **Cache and reuse access tokens** for their full one-hour lifetime instead of requesting one per call.
- **Follow pagination links** rather than assuming a page count.
- **Handle empty collections as normal.** A machine with no data for a period is expected behaviour, not an error condition.
- **Honour `Retry-After`.** Implement exponential backoff on `403` and `429` responses.
- **Split long histories into 14-day windows** when backfilling, and space the requests out so the daily quota is not exhausted in one burst.
- **Store the machine VIN as your key.** It is the identifier every single-machine endpoint requires.
- **Treat cumulative values as running totals.** Subtract consecutive readings to derive period usage; never assume a reading represents a single period.

4. API Reference

This section documents every supported endpoint: its purpose, path, parameters, an example request and a representative response.

Conventions

Base URL

All endpoint paths in this section are relative to the following base URL:

```
https://temeda.azure-api.net/aemp/v2
```

The base URL is confirmed when your subscription is provisioned. Use the hostname exactly as issued.

Required headers

Every data request carries the following headers:

Header	Value	Purpose
`Authorization`	`Bearer {access_token}`	Authorizes the account. The `Bearer` prefix is required.
`Ocp-Apim-Subscription-Key`	`{subscription_key}`	Identifies the subscription to the gateway.
`Accept`	`application/json` or `application/xml`	Selects the response format.

Response samples

> **ABOUT THE SAMPLES** The response samples in this section are shown in JSON and are **representative of the payload structure** for each endpoint — field names, nesting and data types. Values are illustrative. Requesting `application/xml` returns the same data in XML form. For live, byte-exact samples, use the interactive reference on the API portal.

4.1 Fleet Snapshot

GET	fleet/{pageNumber}
Scope	Entire fleet

Returns a snapshot of every machine in scope for your account, with the latest known value for each supported data point. This is the primary endpoint for most integrations — a single call retrieves current location, hour meters, and fuel figures for the whole fleet.

Parameters

Parameter	Type	Required	Description
pageNumber	Integer	Required	The page of results to return. Always supply a value — pass 1 for the first page. Results are returned 100 machines per page.

Example request

```
GET https://temeda.azure-api.net/aemp/v2/fleet/1

Accept: application/json
Authorization: Bearer {access_token}
Ocp-Apim-Subscription-Key: {subscription_key}
```

Example response


```
{
  "Fleet": {
    "Version": "2.0",
    "SnapshotTime": "2026-07-28T14:05:00Z",
    "Equipment": [
      {
        "EquipmentHeader": {
          "OEMName": "Yanmar",
          "Model": "ViO35-6A",
          "EquipmentID": "YARD-1042",
          "SerialNumber": "YMRVIO356A00012345",
          "PIN": "YMRVIO356A00012345",
          "UnitInstallDateTime": "2025-03-14T00:00:00Z"
        },
        "Location": {
          "Datetime": "2026-07-28T13:52:11Z",
          "Latitude": 32.901244,
          "Longitude": -96.829107,
          "Altitude": { "Value": 152.4, "UnitOfMeasure": "meter" }
        },
        "CumulativeOperatingHours": {
          "Datetime": "2026-07-28T13:52:11Z",
          "Hour": 1284.6
        },
        "CumulativeIdleHours": {
          "Datetime": "2026-07-28T13:52:11Z",
          "Hour": 212.3
        },
        "FuelUsed": {
          "Datetime": "2026-07-28T13:52:11Z",
          "FuelConsumed": 3421.7,
          "FuelUnits": "litre"
        },
        "FuelUsedLast24": {
```

```
          "Datetime": "2026-07-28T13:52:11Z",
          "FuelConsumed": 18.4,
          "FuelUnits": "litre"
        },
        "FuelRemaining": {
          "Datetime": "2026-07-28T13:52:11Z",
          "Percent": 62.0
        },
        "DEFRemaining": {
          "Datetime": "2026-07-28T13:52:11Z",
          "Percent": 78.0
        },
        "EngineStatus": {
          "Datetime": "2026-07-28T13:52:11Z",
          "Running": false
        },
        "MaximumSpeedLast24": {
          "Datetime": "2026-07-28T13:52:11Z",
          "SpeedUnit": “km/h”,
          "Speed": 41.0
        }
    ],
    "Links": [
      { "Rel": "self", "Href": "https://temeda.azure-api.net/aemp/v2/fleet/1" },
      { "Rel": "next", "Href": "https://temeda.azure-api.net/aemp/v2/fleet/2" },
      { "Rel": "last", "Href": "https://temeda.azure-api.net/aemp/v2/fleet/12" }
    ]
  }
}
```


Notes

- Paginated at 100 machines per page. Use the `Links` collection in the response to walk to the next, previous or last page.
- Responses are cached for 15 minutes. Calling this endpoint more frequently than that returns the cached payload.
- Values reflect the most recent reported data for each machine. A field is omitted or empty when the machine has never reported that data point.

4.2 Equipment Snapshot

GET	fleet/equipment/{equipmentIdentifier}
Scope	Single machine

Returns the same snapshot payload as Fleet Snapshot, for one machine identified by VIN. Use this when you need current data for a specific machine rather than pulling the whole fleet.

Parameters

Parameter	Type	Required	Description
equipmentIdentifier	String	Required	The machine VIN. This is the asset VIN — not the GPS device serial number.

Example request

```
GET https://temeda.azure-api.net/aemp/v2/fleet/equipment/YMRVIO356A00012345

Accept: application/json
Authorization: Bearer {access_token}
Ocp-Apim-Subscription-Key: {subscription_key}
```

Example response

```
{
  "Fleet": {
    "Version": "2.0",
    "SnapshotTime": "2026-07-28T14:05:00Z",
    "Equipment": [
      {
        "EquipmentHeader": {
          "OEMName": "Yanmar",
          "Model": "ViO35-6A",
          "EquipmentID": "YARD-1042",
          "SerialNumber": "YMRVIO356A00012345",
          "PIN": "YMRVIO356A00012345"
        },
        "Location": {
          "Datetime": "2026-07-28T13:52:11Z",
          "Latitude": 32.901244,
          "Longitude": -96.829107,
          "Altitude": { "Value": 152.4, "UnitOfMeasure": "meter" }
        },
        "CumulativeOperatingHours": {
          "Datetime": "2026-07-28T13:52:11Z",
          "Hour": 1284.6
        },
        "EngineStatus": {
          "Datetime": "2026-07-28T13:52:11Z",
          "Running": false
        }
      }
    ]
  }
}
```

Notes

- No pagination — a single machine is always returned in one response.
- The payload structure is identical to Fleet Snapshot, containing one `Equipment` entry.
- Responses are cached for 15 minutes.

4.3 Time Series — Cumulative Operating Hours

GET	`fleet/equipment/{equipmentIdentifier}/CumulativeOperatingHours/{startDateUtc}/{endDateUtc}/{pageNumber}`
Scope	Single machine, time series

Returns cumulative operating (engine) hours readings for one machine across the requested window, with the UTC timestamp of each reading. Useful for reconstructing hour-meter progression for billing, warranty or utilization analysis.

Parameters

Parameter	Type	Required	Description
`equipmentIdentifier`	String	Required	The machine VIN. This is the asset VIN — not the GPS device serial number.
`startDateUtc`	DateTime (UTC)	Required	Start of the requested window, in UTC. Includes both date and time, e.g. `2026-07-01T00:00:00Z`.
`endDateUtc`	DateTime (UTC)	Required	End of the requested window, in UTC. Must be no more than 14 days after `startDateUtc`.
`pageNumber`	Integer	Required	The page of results to return. Always supply a value — pass `1` for the first page.

Example request

```
GET https://temeda.azure-
api.net/aemp/v2/fleet/equipment/YMRVIO356A00012345/CumulativeOperatingHours/2026-07-
01T00:00:00Z/2026-07-14T23:59:59Z/1

Accept: application/json
Authorization: Bearer {access_token}
Ocp-Apim-Subscription-Key: {subscription_key}
```

Example response

```
{
  "EquipmentIdentifier": "YMRVIO356A00012345",
  "CumulativeOperatingHours": [
    { "Datetime": "2026-07-01T06:12:00Z", "Hour": 1240.1 },
    { "Datetime": "2026-07-02T06:15:00Z", "Hour": 1247.9 },
    { "Datetime": "2026-07-03T06:11:00Z", "Hour": 1255.4 }
  ],
  "Links": [
    { "Rel": "self", "Href": ".../CumulativeOperatingHours/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/1" },
    { "Rel": "next", "Href": ".../CumulativeOperatingHours/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/2" }
  ]
}
```

Notes

- Readings are cumulative lifetime totals, not per-period values. Subtract consecutive readings to derive hours worked between two points.

4.4 Time Series — Cumulative Idle Hours

GET	`fleet/equipment/{equipmentIdentifier}/CumulativeIdleHours/{startDateUtc}/{endDateUtc}/{pageNumber}`
Scope	Single machine, time series

Returns cumulative idle hours readings for one machine across the requested window, with the UTC timestamp of each reading.

Parameters

Parameter	Type	Required	Description
`equipmentIdentifier`	String	Required	The machine VIN. This is the asset VIN — not the GPS device serial number.
`startDateUtc`	DateTime (UTC)	Required	Start of the requested window, in UTC. Includes both date and time, e.g. `2026-07-01T00:00:00Z`.
`endDateUtc`	DateTime (UTC)	Required	End of the requested window, in UTC. Must be no more than 14 days after `startDateUtc`.
`pageNumber`	Integer	Required	The page of results to return. Always supply a value — pass 1 for the first page.

Example request

```
GET https://temeda.azure-
api.net/aemp/v2/fleet/equipment/YMRVIO356A00012345/CumulativeIdleHours/2026-07-01T00:00:00Z/2026-
07-14T23:59:59Z/1

Accept: application/json
Authorization: Bearer {access_token}
Ocp-Apim-Subscription-Key: {subscription_key}
```

Example response

```
{
  "EquipmentIdentifier": "YMRVIO356A00012345",
  "CumulativeIdleHours": [
    { "Datetime": "2026-07-01T06:12:00Z", "Hour": 205.8 },
    { "Datetime": "2026-07-02T06:15:00Z", "Hour": 207.2 }
  ],
  "Links": [
    { "Rel": "self", "Href": ".../CumulativeIdleHours/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/1" }
  ]
}
```

Notes

- Idle hours must be calculated on the machine itself and reported to the platform. Availability therefore varies by machine model and configuration — see *Machine-Data Compatibility*.
- The endpoint returns an empty result set for machines that do not report idle hours.

4.5 Time Series — Cumulative Fuel Used

GET	`fleet/equipment/{equipmentIdentifier}/CumulativeFuelUsed/{startDateUtc}/{endDateUtc}/{pageNumber}`
Scope	Single machine, time series

Returns cumulative fuel used readings for one machine across the requested window, with the unit of measure and the UTC timestamp of each reading.

Parameters

Parameter	Type	Required	Description
`equipmentIdentifier`	String	Required	The machine VIN. This is the asset VIN — not the GPS device serial number.
`startDateUtc`	DateTime (UTC)	Required	Start of the requested window, in UTC. Includes both date and time, e.g. `2026-07-01T00:00:00Z`.
`endDateUtc`	DateTime (UTC)	Required	End of the requested window, in UTC. Must be no more than 14 days after `startDateUtc`.
`pageNumber`	Integer	Required	The page of results to return. Always supply a value — pass 1 for the first page.

Example request

```
GET https://temeda.azure-
api.net/aemp/v2/fleet/equipment/YMRVIO356A00012345/CumulativeFuelUsed/2026-07-01T00:00:00Z/2026-
07-14T23:59:59Z/1

Accept: application/json
Authorization: Bearer {access_token}
Ocp-Apim-Subscription-Key: {subscription_key}
```

Example response

```
{
  "EquipmentIdentifier": "YMRVIO356A00012345",
  "FuelUsed": [
    { "Datetime": "2026-07-01T06:12:00Z", "FuelConsumed": 3298.4, "FuelUnits": "litre" },
    { "Datetime": "2026-07-02T06:15:00Z", "FuelConsumed": 3316.8, "FuelUnits": "litre" }
  ],
  "Links": [
    { "Rel": "self", "Href": ".../CumulativeFuelUsed/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/1" }
  ]
}
```

Notes

- Readings are cumulative lifetime totals. Subtract consecutive readings to derive fuel consumed between two points.

4.6 Time Series — Fuel Used In The Preceding 24 Hours

GET	`fleet/equipment/{equipmentIdentifier}/FuelUsedInThePreceding24Hours/{startDateUtc}/{endDateUtc}/{pageNumber}`
Scope	Single machine, time series

Returns a time series of fuel consumed in the preceding 24 hours for one machine. Each data point provides the fuel consumed, the unit of measure and the UTC timestamp.

Parameters

Parameter	Type	Required	Description
`equipmentIdentifier`	String	Required	The machine VIN. This is the asset VIN — not the GPS device serial number.
`startDateUtc`	DateTime (UTC)	Required	Start of the requested window, in UTC. Includes both date and time, e.g. `2026-07-01T00:00:00Z`.
`endDateUtc`	DateTime (UTC)	Required	End of the requested window, in UTC. Must be no more than 14 days after `startDateUtc`.
`pageNumber`	Integer	Required	The page of results to return. Always supply a value — pass 1 for the first page.

Example request

```
GET https://temeda.azure-
api.net/aemp/v2/fleet/equipment/YMRVIO356A00012345/FuelUsedInThePreceding24Hours/2026-07-
01T00:00:00Z/2026-07-14T23:59:59Z/1

Accept: application/json
Authorization: Bearer {access_token}
Ocp-Apim-Subscription-Key: {subscription_key}
```

Example response

```
{
  "EquipmentIdentifier": "YMRVIO356A00012345",
  "FuelUsedLast24": [
    { "Datetime": "2026-07-01T06:12:00Z", "FuelConsumed": 21.6, "FuelUnits": "litre" },
    { "Datetime": "2026-07-02T06:15:00Z", "FuelConsumed": 18.4, "FuelUnits": "litre" }
  ],
  "Links": [
    { "Rel": "self", "Href": ".../FuelUsedInThePreceding24Hours/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/1" }
  ]
}
```

Notes

- Unlike Cumulative Fuel Used, these values are already per-period — no subtraction is required to get daily consumption.

4.7 Time Series — Fuel Remaining Ratio

GET	`fleet/equipment/{equipmentIdentifier}/FuelRemainingRatio/{startDateUtc}/{endDateUtc}/{pageNumber}`
Scope	Single machine, time series

Returns a time series of remaining fuel tank level for one machine, expressed as a percentage, with the UTC timestamp of each reading.

Parameters

Parameter	Type	Required	Description
`equipmentIdentifier`	String	Required	The machine VIN. This is the asset VIN — not the GPS device serial number.
`startDateUtc`	DateTime (UTC)	Required	Start of the requested window, in UTC. Includes both date and time, e.g. `2026-07-01T00:00:00Z`.
`endDateUtc`	DateTime (UTC)	Required	End of the requested window, in UTC. Must be no more than 14 days after `startDateUtc`.
`pageNumber`	Integer	Required	The page of results to return. Always supply a value — pass 1 for the first page.

Example request

```
GET https://temeda.azure-
api.net/aemp/v2/fleet/equipment/YMRVIO356A00012345/FuelRemainingRatio/2026-07-01T00:00:00Z/2026-
07-14T23:59:59Z/1

Accept: application/json
Authorization: Bearer {access_token}
Ocp-Apim-Subscription-Key: {subscription_key}
```

Example response

```
{
  "EquipmentIdentifier": "YMRVIO356A00012345",
  "FuelRemaining": [
    { "Datetime": "2026-07-01T06:12:00Z", "Percent": 84.0 },
    { "Datetime": "2026-07-01T14:40:00Z", "Percent": 61.0 },
    { "Datetime": "2026-07-02T06:15:00Z", "Percent": 96.0 }
  ],
  "Links": [
    { "Rel": "self", "Href": ".../FuelRemainingRatio/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/1" }
  ]
}
```

Notes

- Sharp increases between consecutive readings generally indicate a refuelling event; sharp decreases outside of operating periods may warrant investigation.

4.8 Time Series — DEF Remaining

GET	`fleet/equipment/{equipmentIdentifier}/DEFRemaining/{startDateUtc}/{endDateUtc}/{pageNumber}`
Scope	Single machine, time series

Returns a time series of remaining Diesel Exhaust Fluid (DEF) tank level for one machine, expressed as a percentage, with the UTC timestamp of each reading.

Parameters

Parameter	Type	Required	Description
`equipmentIdentifier`	String	Required	The machine VIN. This is the asset VIN — not the GPS device serial number.
`startDateUtc`	DateTime (UTC)	Required	Start of the requested window, in UTC. Includes both date and time, e.g. `2026-07-01T00:00:00Z`.
`endDateUtc`	DateTime (UTC)	Required	End of the requested window, in UTC. Must be no more than 14 days after `startDateUtc`.
`pageNumber`	Integer	Required	The page of results to return. Always supply a value — pass 1 for the first page.

Example request

```
GET https://temeda.azure-api.net/aemp/v2/fleet/equipment/YMRVIO356A00012345/DEFRemaining/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/1

Accept: application/json
Authorization: Bearer {access_token}
Ocp-Apim-Subscription-Key: {subscription_key}
```

Example response

```
{
  "EquipmentIdentifier": "YMRVIO356A00012345",
  "DEFRemaining": [
    { "Datetime": "2026-07-01T06:12:00Z", "Percent": 81.0 },
    { "Datetime": "2026-07-02T06:15:00Z", "Percent": 78.0 }
  ],
  "Links": [
    { "Rel": "self", "Href": ".../DEFRemaining/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/1" }
  ]
}
```

Notes

- Only applicable to machines equipped with an SCR after-treatment system that reports DEF level. Machines without DEF return an empty result set.

4.9 Time Series — Engine Condition

GET	`fleet/equipment/{equipmentIdentifier}/EngineCondition/{startDateUtc}/{endDateUtc}/{pageNumber}`
Scope	Single machine, time series

Returns a time series of engine condition (engine running status) for one machine, with the UTC timestamp of each reading. Useful for reconstructing exactly when a machine was started and shut down.

Parameters

Parameter	Type	Required	Description
`equipmentIdentifier`	String	Required	The machine VIN. This is the asset VIN — not the GPS device serial number.
`startDateUtc`	DateTime (UTC)	Required	Start of the requested window, in UTC. Includes both date and time, e.g. `2026-07-01T00:00:00Z`.
`endDateUtc`	DateTime (UTC)	Required	End of the requested window, in UTC. Must be no more than 14 days after `startDateUtc`.
`pageNumber`	Integer	Required	The page of results to return. Always supply a value — pass 1 for the first page.

Example request

```
GET https://temeda.azure-api.net/aemp/v2/fleet/equipment/YMRVIO356A00012345/EngineCondition/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/1

Accept: application/json
Authorization: Bearer {access_token}
Ocp-Apim-Subscription-Key: {subscription_key}
```

Example response

```
{
  "EquipmentIdentifier": "YMRVIO356A00012345",
  "EngineStatus": [
    { "Datetime": "2026-07-01T12:04:00Z", "Running": true },
    { "Datetime": "2026-07-01T20:36:00Z", "Running": false },
    { "Datetime": "2026-07-02T12:11:00Z", "Running": true }
  ],
  "Links": [
    { "Rel": "self", "Href": ".../EngineCondition/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/1" }
  ]
}
```

Notes

- Pair with Time Series — Locations to determine both where a machine was and whether it was running at that point.

4.10 Time Series — Fault Codes

GET	`fleet/equipment/{equipmentIdentifier}/Faults/{startDateUtc}/{endDateUtc}/{pageNumber}`
Scope	Single machine, time series

Returns Diagnostic Trouble Codes (DTCs) reported by one machine across the requested window. Each entry provides the SPN, the FMI, an occurrence count, a fault description and the UTC timestamp.

Parameters

Parameter	Type	Required	Description
`equipmentIdentifier`	String	Required	The machine VIN. This is the asset VIN — not the GPS device serial number.
`startDateUtc`	DateTime (UTC)	Required	Start of the requested window, in UTC. Includes both date and time, e.g. `2026-07-01T00:00:00Z`.
`endDateUtc`	DateTime (UTC)	Required	End of the requested window, in UTC. Must be no more than 14 days after `startDateUtc`.
`pageNumber`	Integer	Required	The page of results to return. Always supply a value — pass 1 for the first page.

Example request

```
GET https://temeda.azure-api.net/aemp/v2/fleet/equipment/YMRVIO356A00012345/Faults/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/1

Accept: application/json
Authorization: Bearer {access_token}
Ocp-Apim-Subscription-Key: {subscription_key}
```

Example response

```
{
  "EquipmentIdentifier": "YMRVIO356A00012345",
  "FaultCodes": [
    {
      "Datetime": "2026-07-03T09:14:02Z",
      "CodeIdentifier": "3667-3",
      "CodeDescription": "Engine Air Shutoff Status"
    },
    {
      "Datetime": "2026-07-09T15:41:37Z",
      "CodeIdentifier": "652-5",
      "CodeDescription": "Injector Cylinder #02 - Injector Solenoid Cylinder #2 Circuit - Current Below Normal, or Open Circuit"
    }
  ],
  "Links": [
    { "Rel": "self", "Href": ".../Faults/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/1" }
  ]
}
```

Notes

- Note the path keyword is `Faults`, while the endpoint is named Fault Codes.
- This is the endpoint that supplies DTC data. Where a manufacturer-specific fault description has been supplied for a given SPN/FMI combination, that description is returned in place of the generic J1939 description. Manufacturer-specific descriptions apply only to that manufacturer's machines.
- The CodeIdentifier field combines SPN and FMI into a single string in the format {SPN}-{FMI}. There are no separate fields for SPN, FMI or occurrence count.

4.11 Time Series — Locations

GET	`fleet/equipment/{equipmentIdentifier}/Locations/{startDateUtc}/{endDateUtc}/{pageNumber}`
Scope	Single machine, time series

Returns the latitude, longitude and altitude of every position reported by one machine across the requested window, with the UTC timestamp of each point. In practice this is the machine's history trail in raw form.

Parameters

Parameter	Type	Required	Description
`equipmentIdentifier`	String	Required	The machine VIN. This is the asset VIN — not the GPS device serial number.
`startDateUtc`	DateTime (UTC)	Required	Start of the requested window, in UTC. Includes both date and time, e.g. `2026-07-01T00:00:00Z`.
`endDateUtc`	DateTime (UTC)	Required	End of the requested window, in UTC. Must be no more than 14 days after `startDateUtc`.
`pageNumber`	Integer	Required	The page of results to return. Always supply a value — pass 1 for the first page.

Example request

```
GET https://temeda.azure-api.net/aemp/v2/fleet/equipment/YMRVIO356A00012345/Locations/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/1

Accept: application/json
Authorization: Bearer {access_token}
Ocp-Apim-Subscription-Key: {subscription_key}
```

Example response

```
{
  "EquipmentIdentifier": "YMRVIO356A00012345",
  "Locations": [
    {
      "Datetime": "2026-07-01T12:04:00Z",
      "Latitude": 32.901244,
      "Longitude": -96.829107
    },
    {
      "Datetime": "2026-07-01T12:19:00Z",
      "Latitude": 32.903810,
      "Longitude": -96.831522
    }
  ],
  "Links": [
    { "Rel": "self", "Href": ".../Locations/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/1" },
    { "Rel": "next", "Href": ".../Locations/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/2" },
    { "Rel": "last", "Href": ".../Locations/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/5" }
  ]
}
```

Notes

- This endpoint typically returns the largest result sets of any time series call. A busy machine over a full 14-day window can span many pages — always follow the `Links` collection until no `next` link is returned.

4.12 Time Series — Average Daily Engine Load Factors

GET	`fleet/equipment/{equipmentIdentifier}/AverageDailyEngineLoadFactors/{startDateUtc}/{endDateUtc}/{pageNumber}`
Scope	Single machine, time series

Returns a time series of average daily engine load factor for one machine, expressed as a percentage, with the UTC timestamp of each reading. Load factor indicates how hard the engine was working relative to its capability.

Parameters

Parameter	Type	Required	Description
`equipmentIdentifier`	String	Required	The machine VIN. This is the asset VIN — not the GPS device serial number.
`startDateUtc`	DateTime (UTC)	Required	Start of the requested window, in UTC. Includes both date and time, e.g. `2026-07-01T00:00:00Z`.
`endDateUtc`	DateTime (UTC)	Required	End of the requested window, in UTC. Must be no more than 14 days after `startDateUtc`.
`pageNumber`	Integer	Required	The page of results to return. Always supply a value — pass `1` for the first page.

Example request

```
GET https://temeda.azure-
api.net/aemp/v2/fleet/equipment/YMRVIO356A00012345/AverageDailyEngineLoadFactors/2026-07-
01T00:00:00Z/2026-07-14T23:59:59Z/1

Accept: application/json
Authorization: Bearer {access_token}
Ocp-Apim-Subscription-Key: {subscription_key}
```

Example response

```
{
  "EquipmentIdentifier": "YMRVIO356A00012345",
  "AverageDailyEngineLoadFactors": [
    { "Datetime": "2026-07-01T00:00:00Z", "Percent": 47.2 },
    { "Datetime": "2026-07-02T00:00:00Z", "Percent": 52.8 }
  ],
  "Links": [
    { "Rel": "self", "Href": ".../AverageDailyEngineLoadFactors/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/1" }
  ]
}
```

Notes

- Because values are averaged per day, this endpoint returns at most one data point per calendar day within the requested window.

4.13 Time Series — Switch Status

GET	`fleet/equipment/{equipmentIdentifier}/SwitchStatus/{startDateUtc}/{endDateUtc}/{pageNumber}`
Scope	Single machine, time series

Returns on/off state changes for the digital switches reported by one machine, with the UTC timestamp of each change. The ISO standard defines a switch generically — it carries a switch number and an on/off state, without describing the switch's purpose.

Parameters

Parameter	Type	Required	Description
`equipmentIdentifier`	String	Required	The machine VIN. This is the asset VIN — not the GPS device serial number.
`startDateUtc`	DateTime (UTC)	Required	Start of the requested window, in UTC. Includes both date and time, e.g. `2026-07-01T00:00:00Z`.
`endDateUtc`	DateTime (UTC)	Required	End of the requested window, in UTC. Must be no more than 14 days after `startDateUtc`.
`pageNumber`	Integer	Required	The page of results to return. Always supply a value — pass 1 for the first page.

Example request

```
GET https://temeda.azure-api.net/aemp/v2/fleet/equipment/YMRVIO356A00012345/SwitchStatus/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/1

Accept: application/json
Authorization: Bearer {access_token}
Ocp-Apim-Subscription-Key: {subscription_key}
```

Example response

```
{
  "EquipmentIdentifier": "YMRVIO356A00012345",
  "SwitchStatus": [
    { "Datetime": "2026-07-04T22:18:00Z", "SwitchNumber": 1, "Locked": true },
    { "Datetime": "2026-07-05T13:02:00Z", "SwitchNumber": 1, "Locked": false }
  ],
  "Links": [
    { "Rel": "self", "Href": ".../SwitchStatus/2026-07-01T00:00:00Z/2026-07-14T23:59:59Z/1" }
  ]
}
```

Notes

- **Switch 1 is mapped to Engine Restart Lock.** Every lock and unlock of the engine is reported through this endpoint as a state change on switch number 1 — `true` when the engine is locked and `false` when it is unlocked.
- This makes Switch Status the endpoint to use for auditing remote engine lock activity through the API. The same history is also visible in the SmartAssist Remote web and mobile applications.
- Switch numbers other than 1 are not currently mapped to a function on Yanmar machines.

Appendix A — Endpoint Quick Reference

All supported endpoints, relative to the base URL `https://temeda.azure-api.net/aemp/v2`.

Endpoint	Path
Fleet Snapshot	`fleet/{pageNumber}`
Equipment Snapshot	`fleet/equipment/{equipmentIdentifier}`
TS: Cumulative Operating Hours	`fleet/equipment/{equipmentIdentifier}/CumulativeOperatingHours/{startDateUtc}/{endDateUtc}/{pageNumber}`
TS: Cumulative Idle Hours	`fleet/equipment/{equipmentIdentifier}/CumulativeIdleHours/{startDateUtc}/{endDateUtc}/{pageNumber}`
TS: Cumulative Fuel Used	`fleet/equipment/{equipmentIdentifier}/CumulativeFuelUsed/{startDateUtc}/{endDateUtc}/{pageNumber}`
TS: Fuel Used In The Preceding 24 Hours	`fleet/equipment/{equipmentIdentifier}/FuelUsedInThePreceding24Hours/{startDateUtc}/{endDateUtc}/{pageNumber}`
TS: Fuel Remaining Ratio	`fleet/equipment/{equipmentIdentifier}/FuelRemainingRatio/{startDateUtc}/{endDateUtc}/{pageNumber}`
TS: DEF Remaining	`fleet/equipment/{equipmentIdentifier}/DEFRemaining/{startDateUtc}/{endDateUtc}/{pageNumber}`
TS: Engine Condition	`fleet/equipment/{equipmentIdentifier}/EngineCondition/{startDateUtc}/{endDateUtc}/{pageNumber}`
TS: Fault Codes	`fleet/equipment/{equipmentIdentifier}/Faults/{startDateUtc}/{endDateUtc}/{pageNumber}`
TS: Locations	`fleet/equipment/{equipmentIdentifier}/Locations/{startDateUtc}/{endDateUtc}/{pageNumber}`
TS: Average Daily Engine Load Factors	`fleet/equipment/{equipmentIdentifier}/AverageDailyEngineLoadFactors/{startDateUtc}/{endDateUtc}/{pageNumber}`
TS: Switch Status	`fleet/equipment/{equipmentIdentifier}/SwitchStatus/{startDateUtc}/{endDateUtc}/{pageNumber}`

Appendix B — Glossary

Term	Definition
AEMP 2.0	The second-generation standardized telematics data specification for heavy equipment, published as ISO 15143-3.
Cumulative value	A lifetime running total, such as operating hours or fuel used. Period usage is derived by subtracting two readings.
DEF	Diesel Exhaust Fluid, consumed by selective catalytic reduction (SCR) after-treatment systems.
DTC	Diagnostic Trouble Code — a fault reported by a machine's onboard systems, identified by an SPN and FMI pair.
Equipment identifier	The machine VIN, used to address a single machine in the API. Not the GPS device serial number.
FMI	Failure Mode Identifier — the J1939 code describing the nature of a fault (for example, value below normal range).
ISO 15143-3	The international standard governing telematics data interchange for construction machinery.
Load factor	A measure of how hard an engine worked relative to its capability, expressed as a percentage.
Snapshot	The latest known value of each data point for a machine or fleet, as opposed to a history.
SPN	Suspect Parameter Number — the J1939 code identifying which component or parameter a fault relates to.
Subscription key	The credential identifying your subscription to the API gateway; quota and rate limits are counted against it.
Time series	A sequence of timestamped readings of one data type for one machine over a date range.

Appendix C — Support

API access requests — All requests for API access, at every level, are submitted to Yanmar Customer Support. Yanmar reviews the request and coordinates provisioning.

Authentication method questions — If you are unsure whether your account uses Modern Auth or Classic Auth, contact Yanmar Customer Support before beginning development.

Technical issues — For persistent errors, contact Yanmar Customer Support with the request URL, the UTC timestamp of the request, the HTTP status code received and the machine identifier involved, if applicable.

Data availability questions — If an endpoint returns empty results for a machine you expect data from, check the Machine-Data Compatibility table in Section 2 first; many data points depend on what the specific model measures and reports.