Data Provider Integration Guide

Send aircraft data logs to Avistrata · API v1 · 25 October 2026

One API key and one upload URL let your devices send every customer's flight and engine data logs straight into Avistrata, with the owner's consent.

Overview

Avistrata is maintenance and flight operations software from Leesburg Aviation, LLC. It is used by Part 91 owners and flight departments, Part 135 operators, Part 145 repair stations and flight schools. Each flight's data log becomes a Flight Log entry with its engine analysis attached:

- engine and flight times, takeoffs and landings
- fuel burned and per-cylinder CHT and EGT
- limit exceedances and trends
- the GPS track on a map

Why integrate

- **For your customers:** no SD cards or USB sticks to carry to the office. Hours, engine trends and limit alerts reach maintenance after every flight.
- **For you:** your data feeds the maintenance records shops and operators already keep. You don't build or host a maintenance product.
- **Light work:** the owner connects once on a page we host, and your system sends files to one URL. We read your existing file format, so you don't need a new export.

How it works

The owner connects each aircraft once. After that, every log you send lands in Avistrata, and maintenance approves the hours before they count.



Step 1 happens once per aircraft; steps 2 to 4 after every flight. Your key can only send logs, never read records.

Uploaded flights are read in full straight away. Aircraft hours change only when the operator's maintenance staff approve each flight.

Supported file formats

Send the files your system already produces; Avistrata works out what each one is.

Source	Files	Notes
Dynon SkyView / SkyView HDX	User Data Log `.csv`	One export can hold many flights; they're split at each power-on. Rows logged before the GPS has a fix are timed from the session clock.
Garmin G1000 / G1000 NXi / G3X Touch / Perspective	`.csv` data logs (`#airframe_info` header)	One file per power cycle.
J.P. Instruments EDM	`.jpi` / `.dat`	One file can hold many flights; the tail number set in the monitor is read.
Avidyne, AirSync and other CSV logs	`.csv` with a header row	Columns are matched by name: RPM, MAP, fuel flow, oil, CHT and EGT per cylinder, position, speeds, altitude, Hobbs and tach.

- Any of these may be gzip-compressed (`.gz`).
- Units are converted automatically: °C to °F, litres to US gallons, bar or kPa to psi, metres to feet.
- Twin engines and turbines are supported (ITT, torque, Ng, N1, N2).
- A file needs a time on each row: a date and time, or seconds since power-on plus a date and time somewhere in the log.

Step 1: Get set up with Avistrata

Send us four things and we register you as a data provider.

You send us	Example
Your company name and logo (https URL), shown to owners on the Connect page	Example Avionics
A short provider id for links	`example-avionics`
Your return addresses: https URLs owners go back to after connecting (one per environment; a trailing `*` allows everything under it)	`https://portal.example.com/mrotracker/callback`
A technical contact email	api@example.com

We send you back a **provider key** (`mrol_…`), displayed only once when it's created, and sent to you securely.

- Keep the key on your servers only, never in a device, app or web page.
- You can have several keys (for example test and production) and we can revoke any of them at once.
- For testing, we can set up a demo organization with sample aircraft for you.

Step 2: The owner connects their aircraft

The owner approves you once per aircraft on a page we host; you never handle their Avistrata sign-in. From your app or portal, send the owner to:

```
https://login.mrotracker.com/connect/logs?provider=example-avionics
  &redirect_uri=https%3A%2F%2Fportal.example.com%2Fmrotracker%2Fcallback
  &state=8f3a2c&tail=N123AB&ref=SN1492
```

Parameter	Required	Meaning
`provider`	yes	your provider id
`redirect_uri`	yes	one of your registered return addresses, URL-encoded
`state`	recommended	your own random value, returned unchanged; check it to stop forged callbacks
`tail`	no	tail number to preselect
`ref`	recommended	your device or account id, such as the unit serial number; used to route uploads (Step 3)

The owner then signs in to Avistrata, sees your name and logo and what you can and can't do, ticks the aircraft and clicks **Connect**. Only people who manage maintenance for that organization can approve: admins, the Director of Maintenance, Maintenance Control and mechanics. They come back to your return address:

```
https://portal.example.com/mrotracker/callback?state=8f3a2c
  &links=7c1e0b6a-2f4d-4a51-9b3e-5d8c1f0a2e77
```

- `links` lists one id per connected aircraft.
- If they cancel, you get `?state=8f3a2c&error=access_denied`.
- The owner can disconnect you at any time in the aircraft's Flight Log. Uploads for that aircraft are then refused with `403 link_revoked`.

Step 3: Upload logs

Every aircraft uses the same URL: `POST https://login.mrotracker.com/api/logs/v1/logs` with your key. Send each file as soon as it's available, for example when the aircraft shuts down or the device next connects.

```
curl -X POST https://login.mrotracker.com/api/logs/v1/logs \
  -H "Authorization: Bearer mrol_YOUR_KEY" \
  -H "X-Device-Ref: SN1492" \
  -H "X-File-Name: 2026-10-20-N123AB-SN1492-16.0-USER_LOG_DATA.csv" \
  -H "Content-Type: application/gzip" \
  --data-binary @2026-10-20-N123AB-SN1492-16.0-USER_LOG_DATA.csv.gz
```

Avistrata finds the aircraft among those connected to you, in this order:

1. `X-Device-Ref`: the `ref` from the Connect page (for example the unit serial number).
2. `X-Tail-Number`.
3. The tail number in the file name or in a J.P. Instruments file.

Request	
Body	the raw file (`text/csv`, `application/octet-stream` or `application/gzip`), or `multipart/form-data` with up to 10 files
`X-File-Name`	the original file name (recommended: it helps us identify the source and airports)
Size	about 6 MB per request; gzip CSV logs, which shrink about 10× (up to 60 MB unpacked)

A successful upload returns `201`:

```
{ "data": { "upload_id": "3b9d…", "status": "imported", "duplicates": 0,
  "flights": [ { "engine_data_flight_id": "a41c…",
                 "start": "2026-10-20T14:00:00Z", "end": "2026-10-20T14:59:59Z",
                 "origin": "KJYO", "destination": "KHGR",
                 "engine_hours": 1.0, "flight_hours": 0.83,
                 "fuel_burned_gal": 8.8, "alert_level": "none" } ] } }
```

`status`	Meaning
`imported`	new flights were added
`duplicate`	every flight in the file is already in Avistrata; safe to ignore
`failed`	the file couldn't be read; `error` says why (the response is `422`)

- **Retrying is safe.** The same file, or the same flight inside a later or longer export, is recognised and not counted twice.
- **Already know the link id?** You can also upload to `POST /api/logs/v1/links/{link_id}/logs`.

Other calls and errors

A few read calls let you keep your list of connected aircraft and upload results in step with Avistrata.

Call	Returns
`GET /api/logs/v1`	your provider name and how many aircraft are connected
`GET /api/logs/v1/links?status=active` **(or** `revoked`**,** `all`**)**	connected aircraft: `id`, `tail_no`, `external_ref`, `status`, `connected_at`, `last_upload_at`
`GET /api/logs/v1/uploads/{upload_id}`	the result of an earlier upload
`GET /api/logs/v1/openapi.json`	the OpenAPI 3.1 description (no key needed)

Every error comes back as `{ "error": { "code": "…", "message": "…" } }`:

HTTP	`code`	What to do
401	`unauthorized`, `key_revoked`	check the key; ask us for a new one
403	`provider_suspended`	contact us
403	`link_revoked`	the owner disconnected you; stop sending for that aircraft
404	`aircraft_not_connected`, `no_links`	the aircraft isn't connected yet; send the owner to the Connect page

HTTP	`code`	What to do
409	`ambiguous`	more than one connected aircraft matches; send `X-Device-Ref`
422	(in `data.error`)	the file couldn't be read; don't retry the same file
5xx	`server_error`	retry later with back-off

Security, privacy and data ownership

Your key can only send logs for aircraft whose owners connected you, and it can't read anything else in Avistrata.

- **Owner consent:** the owner chooses each aircraft and can disconnect you at any time. Every connection and disconnection is recorded in the organization's audit log.
- **What you can see:** the tail numbers you were connected to, your own `ref` values, and the results of your own uploads. Nothing else: no maintenance records, people, other aircraft or other providers' data.
- **No changes to aircraft hours:** each uploaded flight waits for the operator's maintenance staff to approve its times. Approval adds it to the Flight Log and updates aircraft hours, landings and maintenance due items.
- **Keys:** only a hash of your key is stored. Keys can be rotated or revoked at once, and all traffic is HTTPS only.
- **Data ownership:** the flight data belongs to the aircraft's owner or operator, the same as the rest of their Avistrata records. It isn't shared with other organizations.
- **Wrong-aircraft protection:** an upload can only reach an aircraft connected to you, matched by your device id or its tail number; when more than one could match, it is refused rather than guessed.

Limits and good practice

Send one file per flight (or per export) as soon as it's ready, gzip it, and always include `X-Device-Ref`.

- **Request size:** about 6 MB per request, up to 10 files per request, 60 MB unpacked per file.
- **Retries:** retry on network errors and `5xx` with back-off (for example 1, 5 and 30 minutes). Don't retry a `422`, because the same file will fail again.
- **Duplicates are free:** resending a whole log export is fine, because flights already in Avistrata are skipped.
- **Rows need time:** each row needs a date and time, or seconds since power-on with a date and time somewhere in the log. Logs with neither are refused.
- **Revoked links:** check `GET /links` now and then, or handle `403 link_revoked`, and stop sending for aircraft that were disconnected.
- **Test first:** we give you a demo organization with sample aircraft, so you can try the whole flow before any customer connects.

Integration checklist

Most providers can go from key to first imported flight with a callback page and one upload call.

☐ Send us your name, logo, provider id, return addresses and technical contact (Step 1)
☐ Receive your provider key and demo organization

☐ Add a **Connect to Avistrata** button that opens the Connect page with `ref` set to the device serial (Step 2)

☐ Handle the callback: check `state`, then store the `links` ids against the device or account

☐ Upload each new log to `POST /api/logs/v1/logs` with `X-Device-Ref`, gzipped (Step 3)

☐ Handle `201` (`imported` or `duplicate`), `422` (don't retry), `403 link_revoked` (stop sending) and `5xx` (retry)

☐ Test with sample files from your units in the demo organization

☐ Go live and tell your customers

Contact
Jonas Gyllensvaan, Founder, Avistrata · **sales@avistrata.com** · 703.520.5040
Technical questions: **support@avistrata.com API description: login.mrotracker.com/api/logs/v1/openapi.json**