

S3 Layout (Epic 3)

This document defines Nova Cat's S3 bucket strategy and object prefixes.

Goals:

- Separate **private scientific data** from **public site assets**
- Keep raw bytes immutable; track derived artifacts and bundles
- Enable deterministic rebuilds of "full dataset" bundles when new data arrives
- Provide stable object keys that align with `nova_id` and `dataset_id`

Buckets

1) Private data bucket (default)

Bucket: `nova-cat-private-data` (name illustrative)

Contains:

- Raw downloaded bytes (FITS, zips, expanded members)
- Quarantine objects (bytes + metadata snapshots)
- Derived artifacts (plots, normalized representations)
- Bundle artifacts (per-nova full zip + manifest)
- Optional: large workflow payload snapshots (if needed)

2) Public site bucket

Bucket: `nova-cat-public-site` (name illustrative)

Contains:

- Static site releases (versioned)
- Publicly published derived assets (already curated/redacted)

Prefix Conventions (Private Bucket)

All keys are deterministic and **UUID-first**.

Raw spectra

- raw/spectra/<nova_id>/<dataset_id>/primary.fits
- raw/spectra/<nova_id>/<dataset_id>/source.json (small provenance snapshot)
- If downloaded as archive:
 - raw/spectra/<nova_id>/<dataset_id>/archive.zip
 - raw/spectra/<nova_id>/<dataset_id>/unzipped/<relative_path_inside_zip>

Quarantine (bytes and context kept together)

- quarantine/spectra/<nova_id>/<dataset_id>/<quarantine_timestamp>/primary.fits
- quarantine/spectra/<nova_id>/<dataset_id>/<quarantine_timestamp>/context.json
 - include validation status, reason_code, header signature, chosen_profile (if any), and pointers to logs

Normalized / canonical-ish representations (internal)

- derived/spectra/<nova_id>/<dataset_id>/normalized/
 - spectrum.parquet (or spectrum.ecsv , spectrum.fits , etc.)
 - metadata.json (axis units, mapping notes, lossy decisions)
- derived/spectra/<nova_id>/<dataset_id>/plots/preview.png

Photometry uploads (user or upstream ingestion)

- raw/photometry/<nova_id>/<dataset_id>/upload/<original_filename>
- Optional preprocessed outputs:
 - derived/photometry/<nova_id>/<dataset_id>/table.parquet
 - derived/photometry/<nova_id>/<dataset_id>/plots/lightcurve.png

Per-nova "full dataset" bundle (rebuilt only when new data arrives)

- bundles/<nova_id>/full.zip
- bundles/<nova_id>/manifest.json
 - includes bundle_build_id, created_at, list of included dataset_ids and file keys, checksums

Optional: workflow payload snapshots (only if boundary payloads get too big)

- workflow-payloads/<workflow_name>/<job_run_id>/input.json
- workflow-payloads/<workflow_name>/<job_run_id>/output.json

FITS Profile Assets

Profiles are primarily **code/repo artifacts** (recommended):

- docs/specs/spectra-fits-profiles.md is the “Rosetta stone”
- Actual profile logic lives in code alongside provider adapters

If you later need runtime-managed profile assets (e.g., YAML profile definitions), store in private bucket:

- profiles/fits/<profile_id>/<version>/profile.yaml
- profiles/fits/<profile_id>/<version>/tests/<sample>.fits

But default is: **keep profiles in repo** for deterministic deployments and change control.

Prefix Conventions (Public Site Bucket)

Release-based publishing (immutable releases)

- releases/<release_id>/index.html
- releases/<release_id>/assets/...
- releases/<release_id>/nova/<nova_id>/...

Current pointer (optional, via copy or redirect config)

- current/... mirrors latest release

Examples

Spectra FITS:

- `raw/spectra/4e9b0e88-.../2c7d1f4d-.../primary.fits`

Quarantine context:

- `quarantine/spectra/4e9b0e88-.../2c7d1f4d-.../2026-02-21T20:45:10Z/context.json`

Bundle:

- `bundles/4e9b0e88-.../full.zip`
- `bundles/4e9b0e88-.../manifest.json`

Mermaid Diagrams

Bucket overview

