IGS 2026 WORKSHOP
SANTIAGO DE CHILE
33°26' S 70°39' O

1–5 June 2026

Metadata Harmonization of GNSS Data Repositories

Rui Fernandes[1], Fernando Geraldes[1], David Mencin[2], Alex Hamilton[2], Henry Berglund[2], Markus Bradke[3], Nils Brinckmann[3], Brandon Owen[4], Ryan Ruddick[4]

[1] UBI, Portugal

[2] EarthScope, U.S.A.

[3] GFZ, Germany

[4] Geosciences Australia, Australia

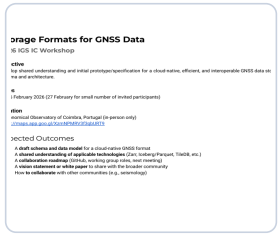
From independent implementations to community common effort

Jan 2026

Abstract submission deadline

Feb 2026

Workshop on storage formats for GNSS data, promoted with active IGS community participation



Workshop

Apr 2026

Whitepaper drafts shared architecture, schemas and API concepts

Whitepaper

31 May 2026

“Data Architecture” IGS Pilot Project approved

Pilot Project

...

IGS 2026 WORKSHOP
SANTIAGO DE CHILE
33°26' S 70°39' O

1-5 June 2026

Part A:
New GNSS Data Formats

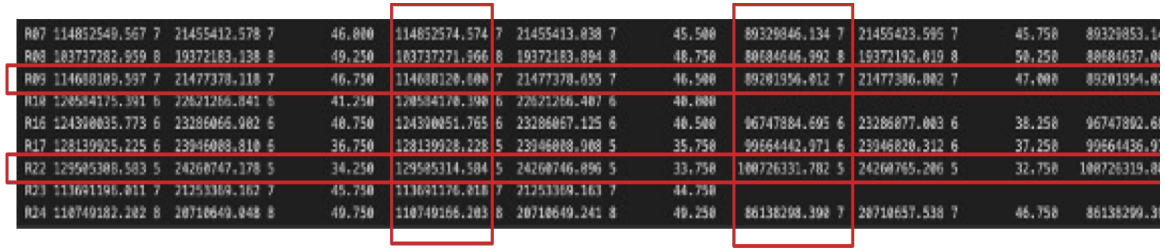
From RINEX files to interoperable GNSS data services

- **Why change?**
  - RINEX remains essential, but file-based archives do not scale for modern discovery, sub-setting and federation.
- **What is changing?**
  - Structured archives using array, columnar or table formats; APIs generate user-facing products on demand.
- **Why metadata?**
  - Federation requires common, queryable information on what each repository contains.
- **What is being built?**
  - Standard data models, availability schemas and APIs for repository discovery, comparison and federated access.

RINEX is the standard, but file archives are now a bottleneck

Main Limitations

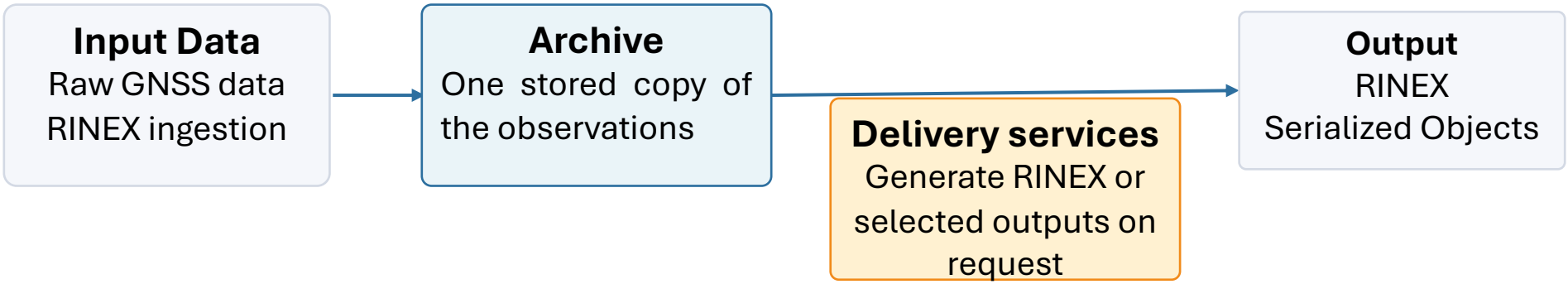
- Access is organized around files, not queryable time windows.
- Subsetting by time, satellite, signal or observable requires reading full files.
- The same measurements are often repeated in separate files at different sampling rates.
- Metadata corrections require regenerating or republishing files.
- Millions of small files are inefficient in cloud/object storage.



- Sometimes I only want certain constellations
- Sometimes I only want certain observables (e.g. phase+range vs SNR)
- Sometimes I want 30 hours of data so I have full satellite arcs on either end of the UTC day

Target model: data-centered archive instead of a file-centered archive

Keep one stored representation of the observations and generate the required files when needed



RINEX remains essential for interoperability, while the storage structure can be better suited to large, sparse GNSS data.

EarthScope operational experience

Example of structured GNSS observation storage and API-based access.

Operational Pattern

- GNSS observations stored in a structured multidimensional data store.
- Station metadata managed separately.
- RINEX 2, 3, and 4 generated on request.
- Access services support selective extraction.
- SDK (Client tools) simplify access for users and applications.
- Architecture optimized for cloud object storage.

```
from earthscope_sdk import EarthScopeClient

es = EarthScopeClient()   ← transparent auth handling

table = es.data.gnss_observations(   ← no URL building
    start_datetime=datetime(2025, 7, 1),   ← Python types
    end_datetime=datetime(2025, 7, 30),
    station_name="AC60",
```

EarthScope SDK: user requests what is needed

```
table = await es.data.gnss_observations(
    start_datetime=dt.datetime(2025, 8, 1),
    end_datetime=dt.datetime(2025, 8, 30),
    station_name="AC60",
    system="G",                    ← just GPS
    obs_code=["1C", "2W"],         ← just signals 1C & 2W
    field=["phase", "range"],      ← just phase & range
```

Server-side subsetting by time, system, code and field

Technology landscape: storage and access layers

Several implementations are being explored; interoperability depends on common schemas (data and metadata) and interfaces.

Storage candidates

TileDB

array storage engine for sparse or dense multidimensional data

Parquet

columnar file format for efficient storage and analytical access

Iceberg

table metadata layer for managing large file-based datasets in object storage

Access / interchange

Arrow

in-memory columnar format for efficient data exchange between storage, APIs, and processing tools

EarthScope is using TileDB operationally, but the community is still evaluating alternative storage approaches.

IGS 2026 WORKSHOP
SANTIAGO DE CHILE
33°26' S 70°39' O

1-5 June 2026

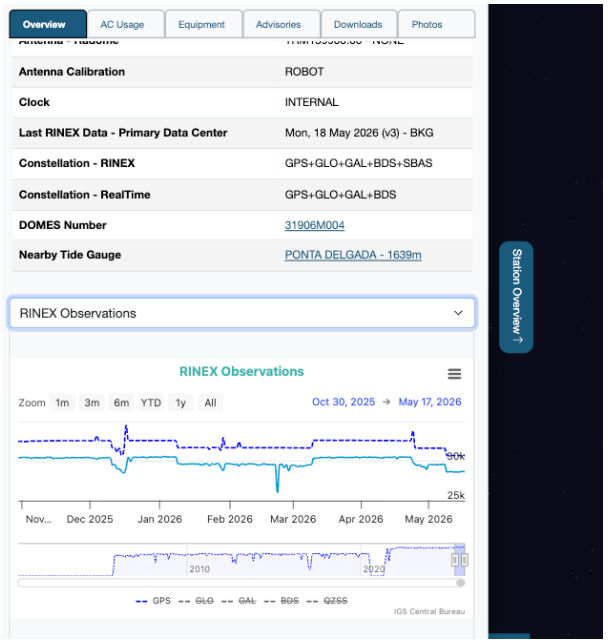
Part B: Metadata Harmonization of GNSS Data Repositories

"What data do you exactly have?"

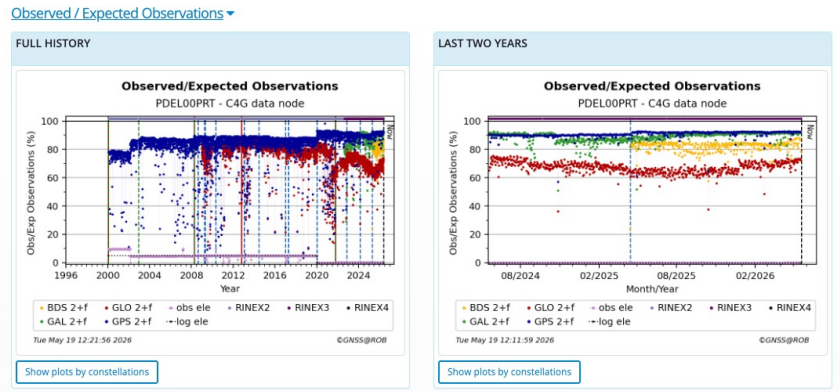
A service must know what each repository contains.

Metadata Attributes

- Station and stream identifiers
- Time windows and sampling
- Constellations, satellites and signals
- Epoch completeness and observation counts
- Integrity information for synchronization



IGS



EPOS

EarthScope

Current metadata representations are not directly comparable

EarthScope and EPOS stores similar information, but through different storage paths and conventions.

EarthScope

Data

- Observation data in array-oriented structures.

Metadata

- Metadata separated from observations.
- Availability derived from array contents and operational indexes.

EPOS

Data

- GNSS data distributed as RINEX files.

Metadata

- Availability and quality information from relational metadata.
- Availability derived from files and QC/database records.

Current metadata representations are not directly comparable

EarthScope

- station_edid
- session_edid
- observation window
- epoch_count / expected_epoch_count
- satellite masks
- signal masks
- system mask
- …

EPOS

- station_id
- reference_date
- Constellation
- Nsat
- epo_expt / epo_have / epo_usbl
- xint_slp
- obstype
- frequency_band
- obs_have / obs_expt / obs_sats
- pha_slps / cod_mpth
- elevation_cutoff
- …

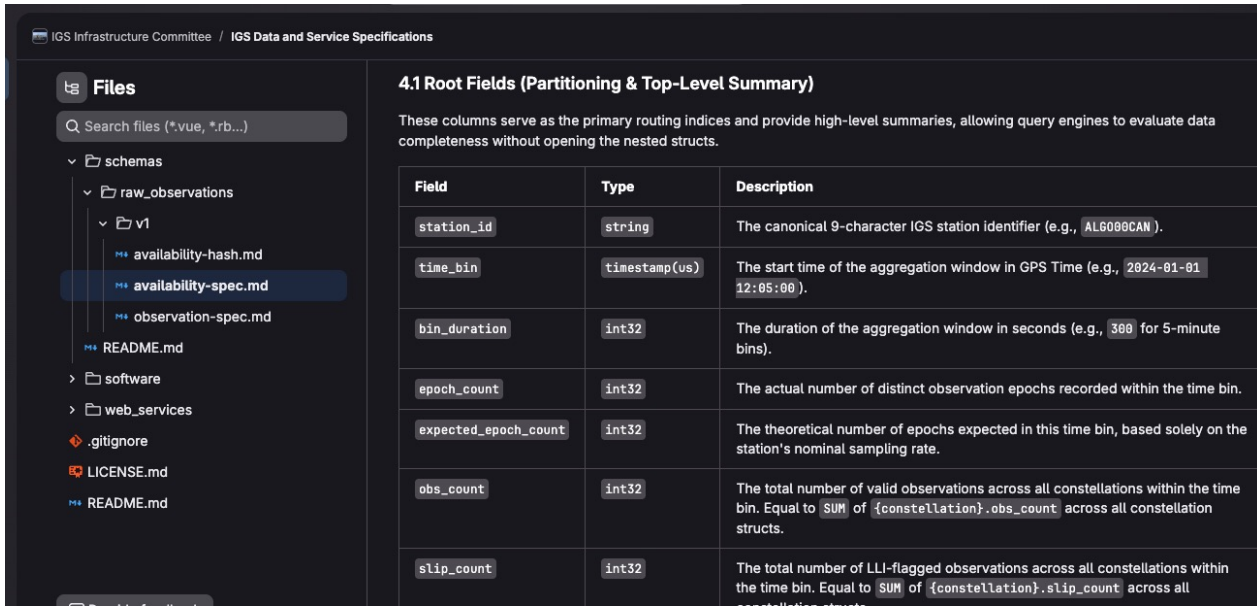
A common schema must define the necessary attributes first;
each repository then maps its current implementation to those attributes.

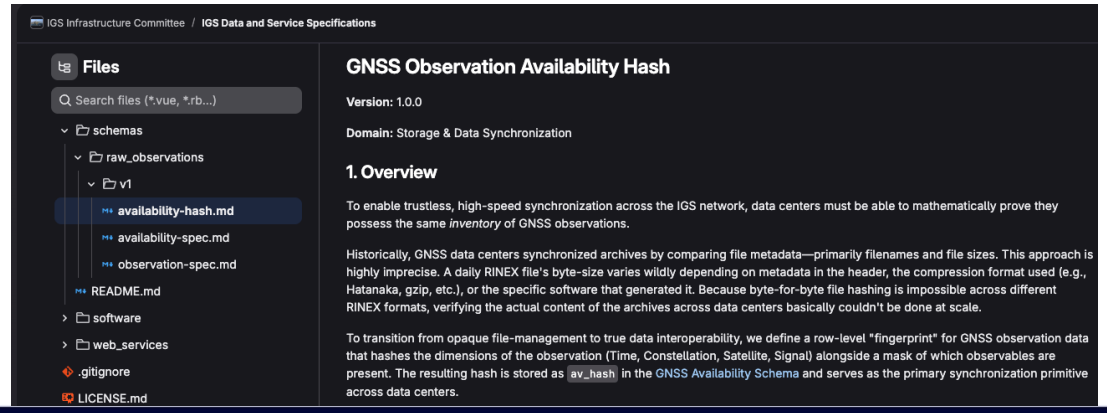
Metadata Schema – IGS community

Minimum information needed to describe holdings in a machine-queryable way.

https://git.gfz-potsdam.de/igs/igs-data-specifications

- The current discussion is centered on which metadata attributes are fundamental and must be defined consistently across repositories.
- Many of these attributes are represented as masks within each time bin, for example the presence of constellations, satellites, signals, and observables.

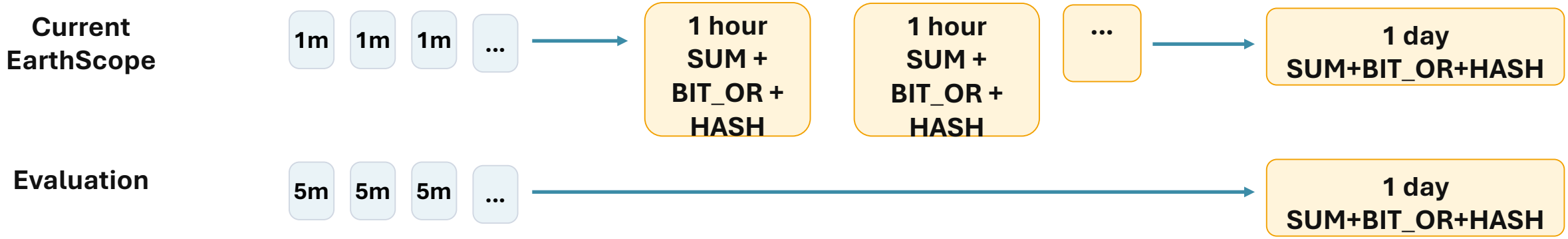




- A key field under discussion is av_hash, which allows fast comparison of observation content between repositories for the same time bin.

Time-Bin & Aggregation (Operationalization)

Based on existing tools made available by EarthScope and tested at EPOS repositories – developed API by UBI



Counts aggregate by SUM:
- epochs
- observations
- LLI / raw flags

Presences aggregate by BIT_OR:
- constellations
- satellites
- signals

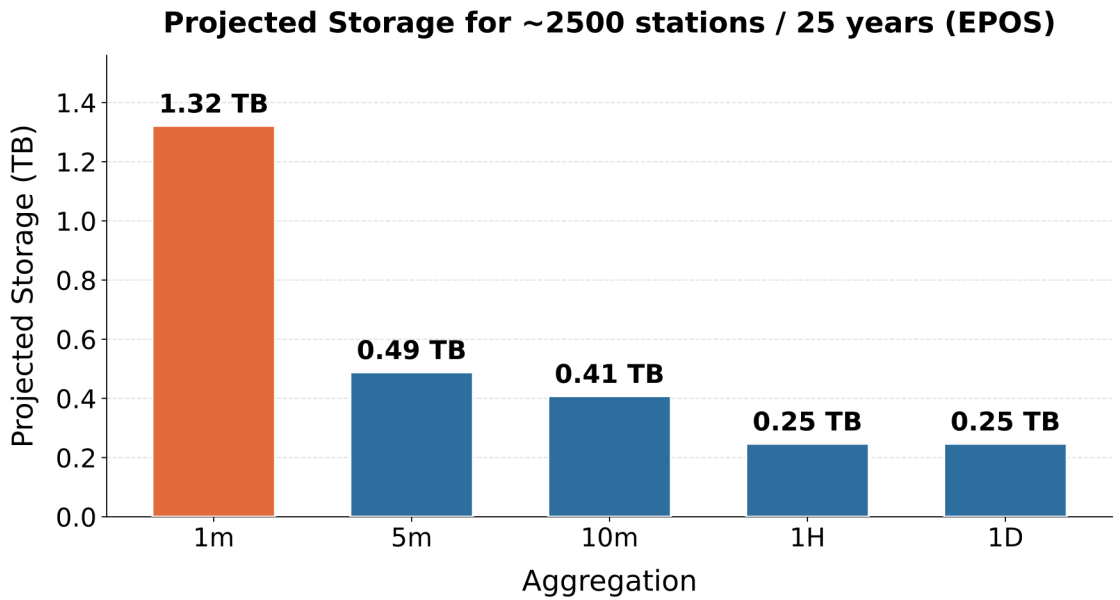
Integrity HASH enables:
- repository comparison

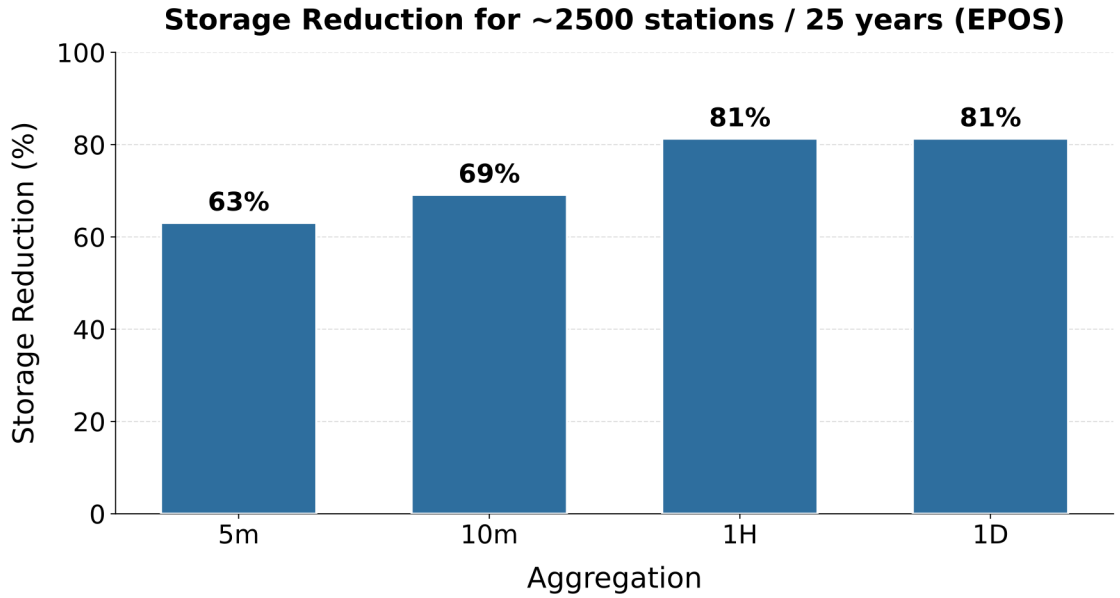
Finer bins increase storage requirements but preserve more detailed availability information

Time-Bin & Aggregation (Storage)

Required Storage of Parquet Files – extrapolation based on the tested UBI repository

UNIT: YEAR -> MONTH -> SITE_DAYOFMONTH





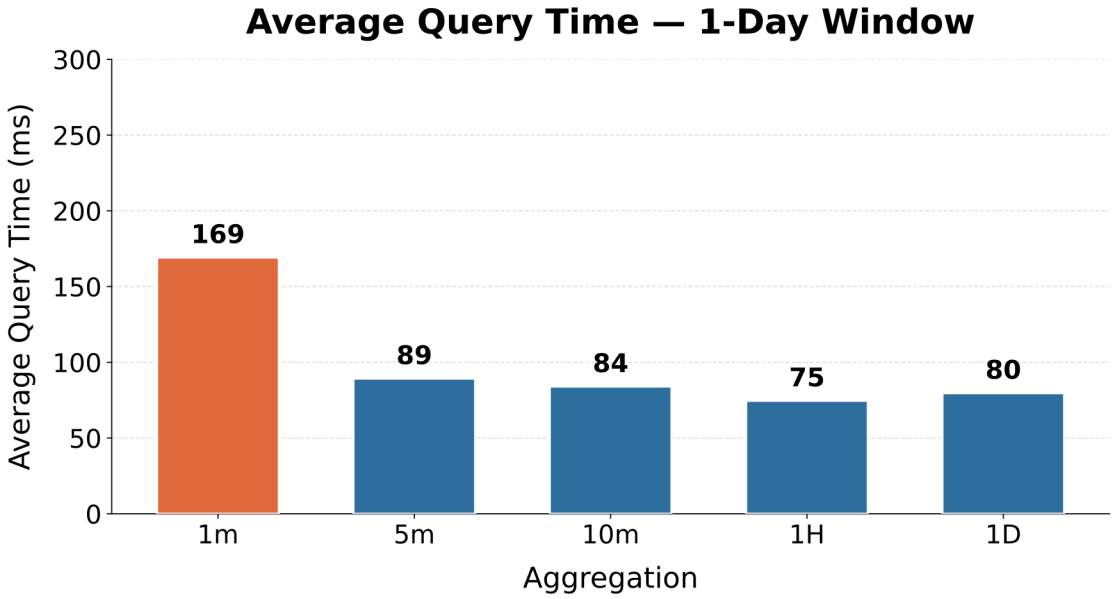
1min time-bin occupies more space than the other time-bins

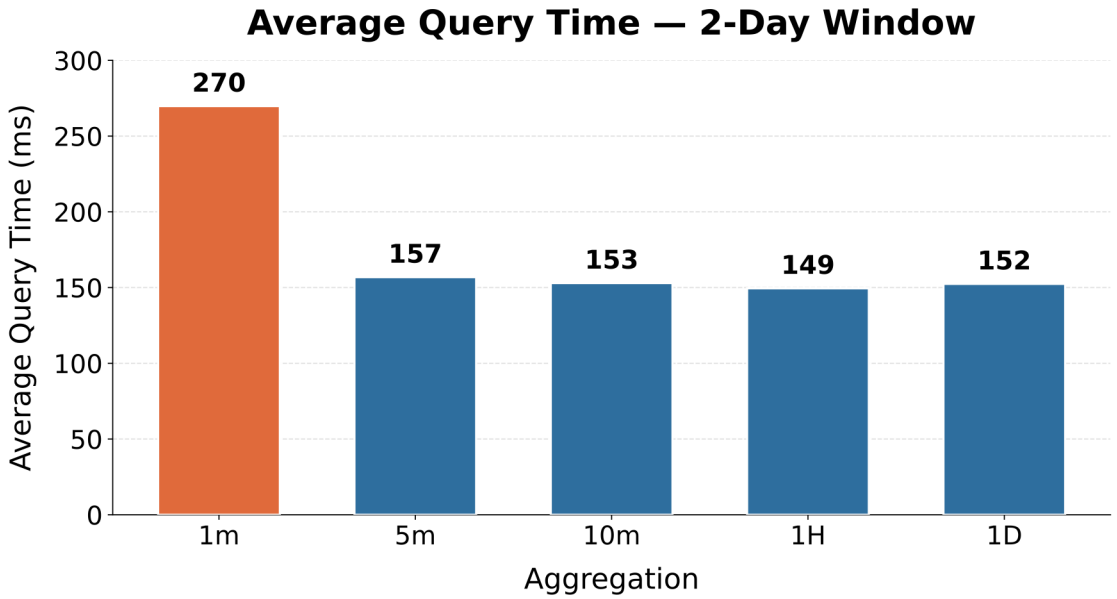
Time-Bin & Aggregation (Access)

Access Time to the Parquet Files – based on tested UBI repository

UNIT: YEAR -> MONTH -> SITE_DAYOFMONTH

Comparing access to one Parquet file or two Parquet files





1min time-bin requires more time than the other time-bins

Conclusions

- GNSS archives are moving from file-centric storage toward structured, queryable data systems.
- RINEX remains essential for exchange and legacy compatibility; the archive and API layer can evolve.
- IGS is an essential body to coordinate the efforts being carried out by many organizations.
- The critical federation layer is not only the observation format; it is standardized discovery and availability metadata.

IGS 2026 WORKSHOP
SANTIAGO DE CHILE
33°26' S 70°39' O

Join us ...

email:
gnss.formats@segal.ubi.pt

There is still (some) room ...