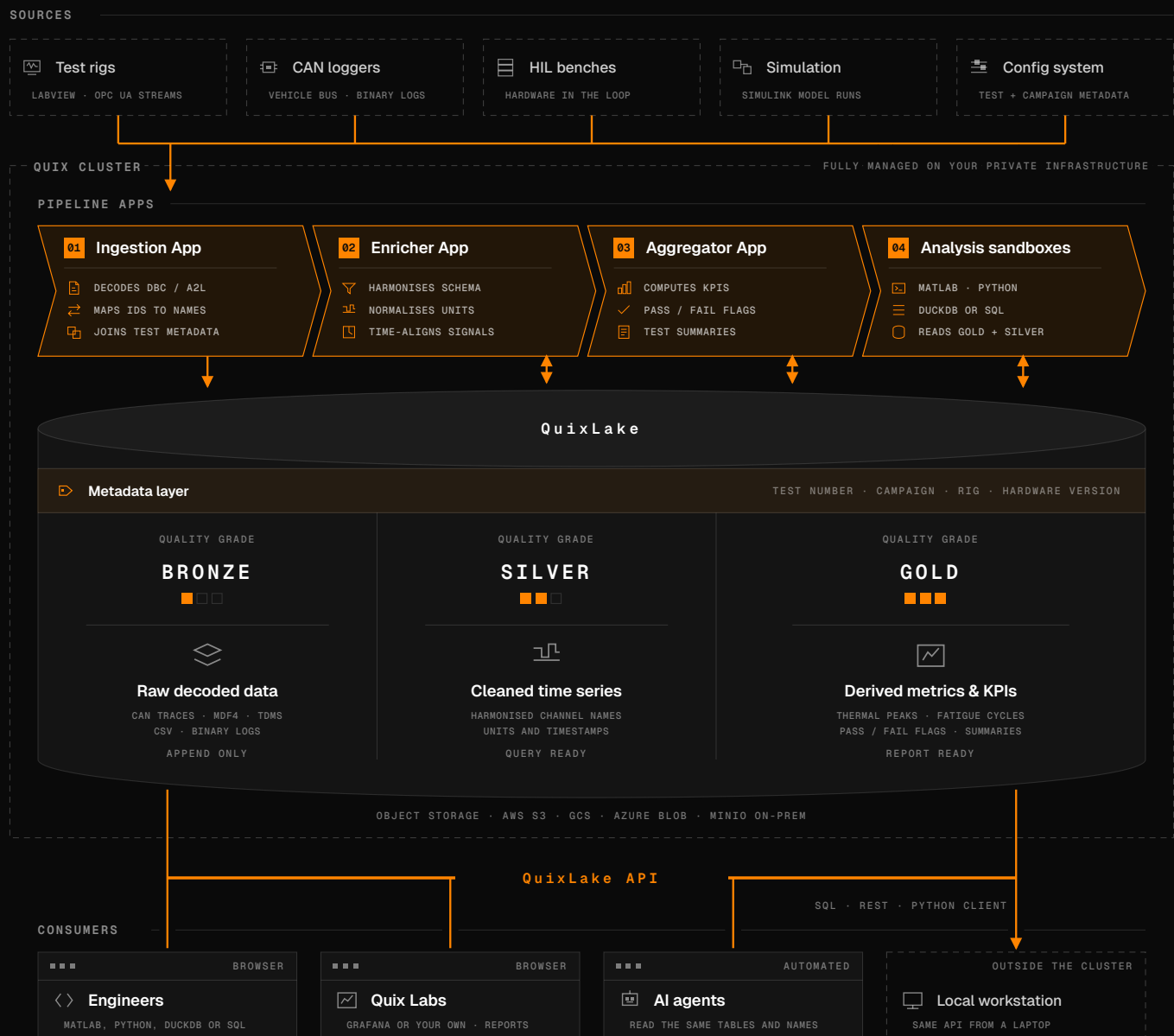


QuixLake

A search engine for full-resolution test data: store it once, find it years later using natural language

Centralize every test recording in your own object storage at full resolution

QuixLake lets your engineers query full-resolution test data in seconds or have AI agents running queries on their behalf. QuixLake is built on object storage which is inexpensive, so keeping terabytes of data will not blow up your IT budget. You can store aggregated views alongside the raw signals in case you want to preprocess them again. Files are written in ordinary formats that common engineering tools can open, so you can still read your data even if you stop using Quix.



Raw test data is expensive to store and hard to manage

A test finishes and the raw data is exported. An engineer copies the MDF files to a shared drive. Before archiving, the channels are reduced to a lower sample rate, because keeping everything is expensive. The report is signed and the original recording is deleted.

Two years later a component fails in service. The measurement that would explain the failure no longer exists — or it survives in a folder only the engineer who ran the test can identify. Over shorter periods the same problem appears: nobody can find last spring's files, so the test is run again.

[01] Raw and transformed data, side by side

QuixLake stores the original recording alongside its harmonized and normalized version, so you can reprocess from source if you later discover a conversion issue such as an incorrect scaling factor or a mismapped channel. You can also publish continuously updated views on the cleaned data — aggregations and derived metrics your team checks regularly — so colleagues do not have to rerun the same queries. This is often called a medallion architecture: three levels of data quality, held together.



BRONZE

Data as recorded. Every channel exactly as the instrument wrote it — nothing corrected, converted or removed. The version you produce when a certification body asks what the equipment actually measured.



SILVER

Units converted to one standard, channel names from different rigs matched to common names, and each run stored with identifying metadata. The version engineers query from day to day.



GOLD

Aggregated views for different stakeholders — a campaign summary with one row per run holding peak temperature, time above a limit, cycle count and pass or fail.

You can maintain each quality level continuously with QuixApps for silver and gold transformations. Whenever new data arrives, it is processed in a Quix Pipeline and your silver and gold tables are automatically updated.

[02] An AI-powered search engine for your test data

QuixLake is tightly integrated with QuixAI, so it knows how to reach your technical documentation and interpret your data. From the chat interface you can start a research project and spawn multiple agents that run analysis pipelines, iterating on different queries until the research goal is met. That only works when test data sits in one place and is standardized enough to compare against other sources such as simulation output — which is why every record is joined with its test configuration on the way in. The test number, campaign, hardware version, operator and ambient conditions are pulled from your configuration system and attached at the moment of ingestion.

Instead of hunting for a file name, an engineer can ask QuixAI to retrieve every run on cooling package revision C where the outlet temperature passed 95 degrees. Nobody has to remember who ran the original test, or where the file was put.

SEE IT WITH YOUR OWN DATA

Book a product demonstration and query full-resolution test data straight from your own cloud storage.

BOOK A DEMO

READ THE DOCUMENTATION

QuixLake overview

Data lake, lakehouse and how they fit together

Lakehouse query service

The SQL surface: catalog, REST API and in-portal editor

Blob storage connections

Bringing your own S3, Azure Blob, GCS or MinIO bucket