houdata.

PRIVATE PREVIEW

PRODUCT DOCUMENTATION

CacheOLAP

Product brief

Reviewed November 2026

CACHEOLAP

Analytical results, ready to reuse

CacheOLAP sits on selected analytical query paths and can reuse eligible results under an explicit freshness policy. Teams can evaluate repeat response time and platform work using their own workload.

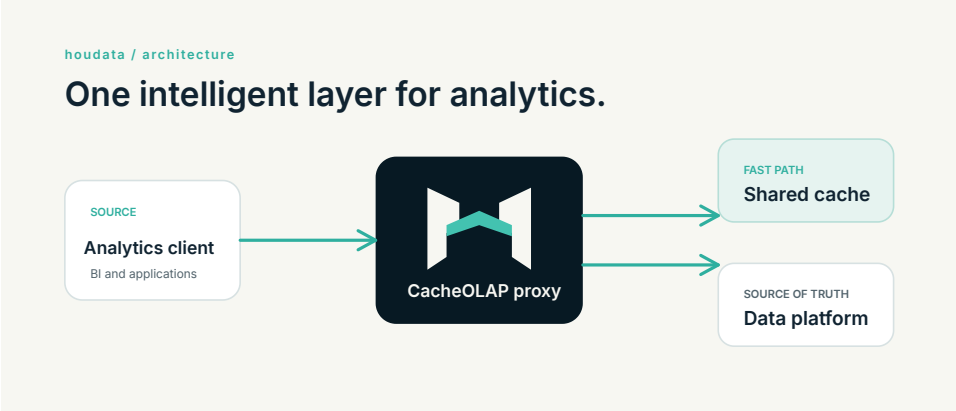
Four platforms, specific paths

The private preview includes experimental integrations for BigQuery, Databricks, Spark and Snowflake. Each follows selected client paths; compatibility depends on the client, configuration and result format. Large results downloaded directly from cloud storage may bypass the proxy.

Policy, identity and freshness

Configurable policies determine which reads are eligible and how long results can be reused. Identity and session context affect whether a result can safely be shared. Analytical data can change through jobs and clients outside the proxy: choose a TTL and clearing strategy around the update cycle.

How the request path works



Illustrative architecture. Each platform has distinct client, identity and result boundaries.

A client connects to the matching proxy using a qualified platform path. A miss reaches the analytical platform; an eligible repeated read can be answered from the shared result cache. Writes and unsupported or incomplete requests continue to the backend. CacheOLAP does not translate SQL between platforms.

Enterprise integration and controls

The integrations account for platform identity and session context when determining reuse. Per-principal isolation and broader sharing have different permission implications and must be reviewed for the deployment. TLS, certificates, cache network isolation and access to platform identity services are part of the deployment design. Exact behavior depends on the integration.

Programmable cache policy

Operators use reviewed scripts and hooks to control eligibility, set lifetimes and bypass selected reads. Observed writes can trigger configured invalidation; data changed by external jobs or uninspected APIs needs a separate freshness strategy. Cache reuse is not a substitute for the platform authorization model or a guarantee of immediate consistency.

Operations and deployment

The Houdata console, shared with CacheDB, shows query activity, cache outcomes and latency, with a Work Saved panel that reports platform work spared in the units the platform reports, never converted to money. Teams can begin by observing traffic, define a bounded policy and compare a representative workload. Sign-in supports OpenID Connect, a Prometheus exporter feeds Grafana, and AI agents can use the console through MCP. Houdata supports deployment in controlled environments, including isolated installation where the necessary services are reachable; cloud platforms still require their own network endpoints.

Suitable workloads and boundaries

Repeated dashboards, reports, aggregations and embedded analytics are candidates when the result can be reused within the permitted freshness window. One-off bulk transfers and results delivered outside the proxy are less suitable. Results vary with repetition, size, identity, freshness and platform billing.

Evaluate with your workload

Select a representative platform, client and recurring query. Agree the freshness and identity boundaries, compare direct and proxied results, then measure hits, latency and platform work. Financial impact depends on the workload and billing model. Contact sales@houdata.eu to discuss a private evaluation.

> **Talk to Houdata**
>
> sales@houdata.eu · +34 630 802 205 · www.houdata.eu. Do not send credentials or production data by email.