ENTERPRISE SEMANTIC SQL ENGINE

ConverseSQL.

SQL Generation Engine · Semantic-Layer Grounded · Embedded across xAQUA

Natural language in. Production SQL out.
The SQL reasoning engine at the core of the xAQUA platform.

ConverseSQL turns plain-English questions into **production-ready SQL** across diverse target data sources. It is grounded by the **xAQUA semantic layer** — not generic LLM guessing — so the SQL it produces uses tables that exist, joins your team has approved, and dialects your engines understand. Proven in production on enterprise schemas with **500+ tables and 50+ columns per table on average**.

NATURAL LANGUAGE PROMPT

"active members enrolled in the COLA-eligible service tier, by employer plan, last 5 fiscal years"

↓

```sql
-- Active COLA-eligible members by employer plan · FY trend  [target: snowflake]
SELECT ep.plan_code, ep.plan_name, fy.fiscal_year,
       COUNT(DISTINCT m.member_id) AS active_members
FROM   dim_member        m
JOIN   dim_service_tier  st ON st.tier_key = m.tier_key
JOIN   dim_employer_plan ep ON ep.plan_key = m.plan_key
JOIN   dim_fiscal_year   fy ON fy.fy_key   = m.enrollment_fy_key
WHERE  st.cola_eligible = TRUE AND m.status = 'ACTIVE'
GROUP BY 1,2,3 ORDER BY 3 DESC;
```

✓ **semantic-resolved** cola_eligible · active_member governance applied · **dialect: snowflake**

© 2026 XFUSION TECHNOLOGIES

WHY CONVERSESQL EXISTS

Generic AI doesn't know your schema.

Real enterprise data does not live in textbook 3NF. It lives in **500+ table warehouses**, wide fact tables with 50+ columns, decades of legacy naming, multiple SQL dialects, and dozens of governance rules nowhere in the DDL. A general-purpose copilot generates SQL that looks right and silently returns the wrong number.

500+	50+	12+	VPC
TABLES / SCHEMA	AVG COLUMNS / TABLE	TARGET SQL DIALECTS	PRIVATE DEPLOY

THE REAL PROBLEM

Four reasons enterprise text-to-SQL usually fails.

01 · SCALE

Too many tables to fit in a prompt

A 500-table schema doesn't fit in any context window. Naive RAG retrieval misses the joins. ConverseSQL resolves the right slice via the semantic graph, every time.

02 · WIDTH

Wide tables with cryptic columns

50+ columns per table, half of them flag_27 or amt_lcl_v2. Without business meaning attached, the model picks the wrong column. The semantic layer makes meaning explicit.

03 · DIALECT

12 dialects, one prompt

Snowflake date math, BigQuery arrays, Postgres JSON, Oracle's outer-join syntax. Generated SQL must be valid in the target — not a generic dialect that needs rewriting.

04 · GOVERNANCE

Rules not in the DDL

"Active member" is a 3-condition filter. "Net revenue" excludes specific channels. These are policy, not schema. ConverseSQL inherits them from the semantic layer, automatically.

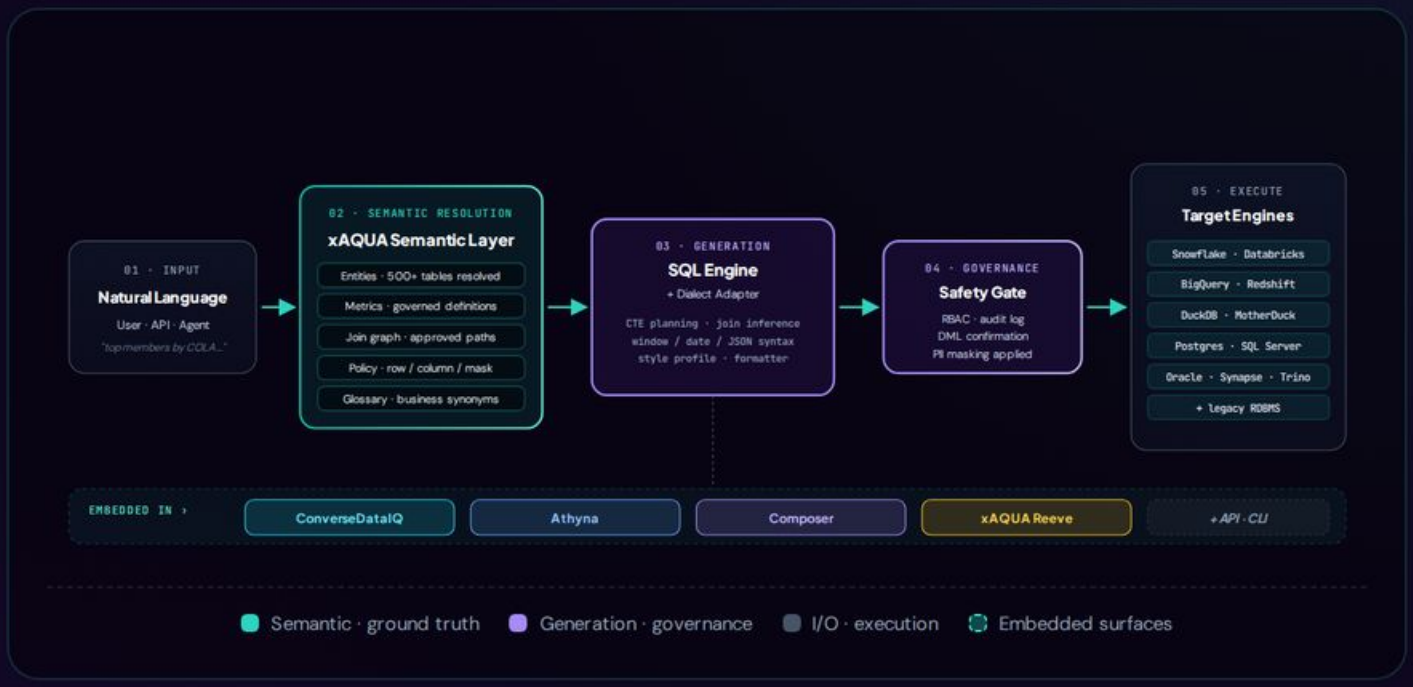
Resolved before generated. Governed before executed. **Production from day one.** ConverseSQL is built for the enterprise environments where text-to-SQL has historically failed — wide schemas, many dialects, governance everywhere.

© 2026 XFUSION TECHNOLOGIES

HOW CONVERSESQL WORKS

Five stages, one engine.

The semantic layer does the heavy lifting before any token of SQL is generated. Resolution happens against the catalog of business entities, governed metrics, join graph, security policies, and dialect mappings — not against a generic prompt-time guess.



ONE INTERFACE · MANY TARGETS

Snowflake	Databricks	BigQuery	DuckDB	MotherDuck	Postgres
CLOUD DW	LAKEHOUSE	CLOUD DW	ADL · NATIVE	SERVERLESS	RDBMS

© 2026 XFUSION TECHNOLOGIES

CORE PLATFORM CAPABILITY

Embedded across xAQUA.

ConverseSQL is not a side feature. It is the SQL generation engine the rest of the platform depends on — wherever a user, agent, or pipeline needs governed SQL from intent.

ConverseDataIQ
Chat with Data

Conversational analytics for business users. Every "chat with data" question routes through ConverseSQL to produce **governed, semantically-grounded answers** from enterprise sources — not LLM guesses.

```
role: NL › governed analytics SQL
```

Athyna
Cloud Data Studio · Prep & Transform

No-code / low-code prep and transformation. Users describe a cleansing rule, a join, a filter — Athyna calls ConverseSQL to **generate the underlying SQL with full pushdown** to the warehouse.

```
role: no-code prep › executable SQL
```

Composer
ETL · Pipelines · Orchestration

Pipeline development environment. ConverseSQL accelerates ETL build, transformation logic, migration translation, and pipeline test generation. **Engineers describe; the engine drafts**; humans review and ship.

```
role: pipeline intent › idempotent SQL
```

xAQUA Reeve
Data Product Catalog

Catalog of certified data products. Reeve uses ConverseSQL for **data product exploration, governed access, and self-serve consumption** — letting consumers query certified products in plain English without breaking governance.

```
role: browse › policy-aware SQL
```

PRODUCTION-SCALE CONTROL

Governance, security, deployment built in.

Private VPC Deployment
Runs entirely inside the customer boundary. Schema, prompts, and SQL never leave. AWS · Azure · GCP · on-prem · air-gap. BYO LLM (GPT-OSS, Llama).

RBAC & Row-Level Security
Generated SQL respects the user's permissions. Won't write a query the user can't run. SSO/SAML/OIDC. PII masking honored automatically.

Full Audit & Lineage
Every prompt, resolved entities, generated SQL, executor, user, and result count logged. Per-user audit trail. Lineage of resolved tables and columns.

Bring your schema. See your SQL.

Watch ConverseSQL generate production SQL against **your** semantic layer, your dialect, your governance rules. Thirty minutes, live against your environment — including the wide tables and the legacy joins.

› Request a Demo | xaqua.io/conversesql | hello@xaqua.io

© 2026 XFUSION TECHNOLOGIES