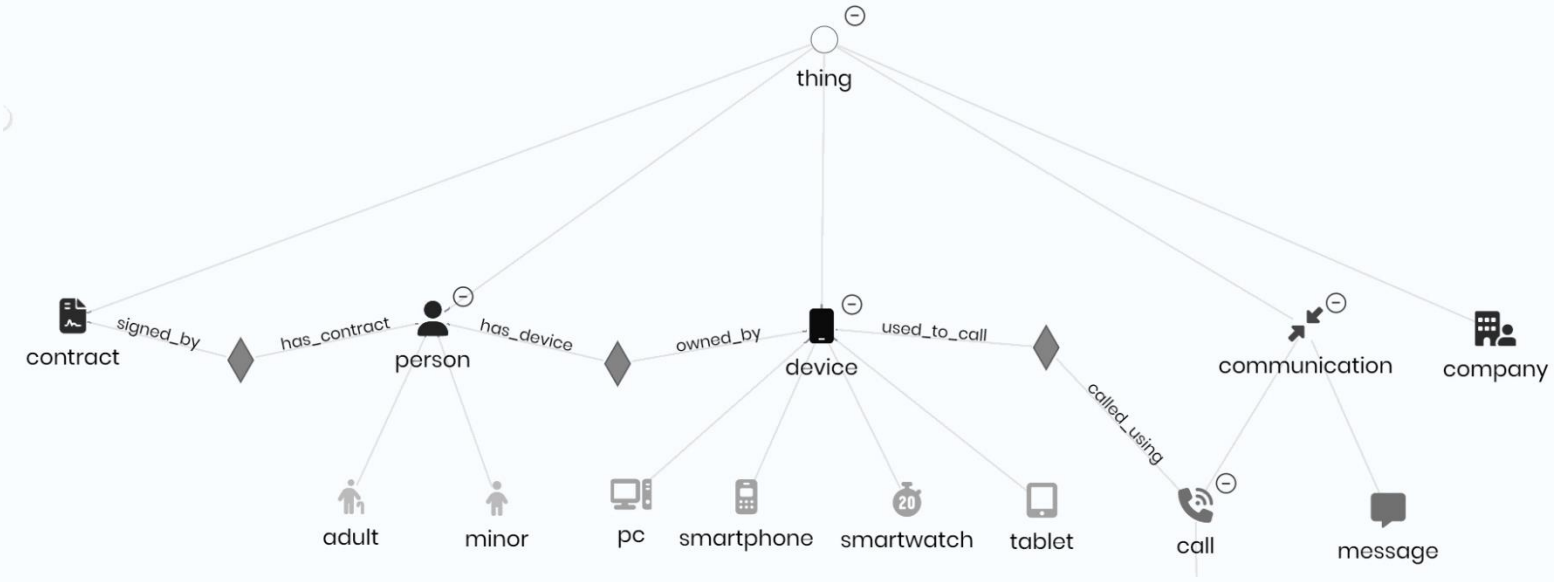
timbr

The Semantic Foundation of the AI Era

SQL-Ontologies Platform for AI, Analytics, and Scalable Data Products and Apps



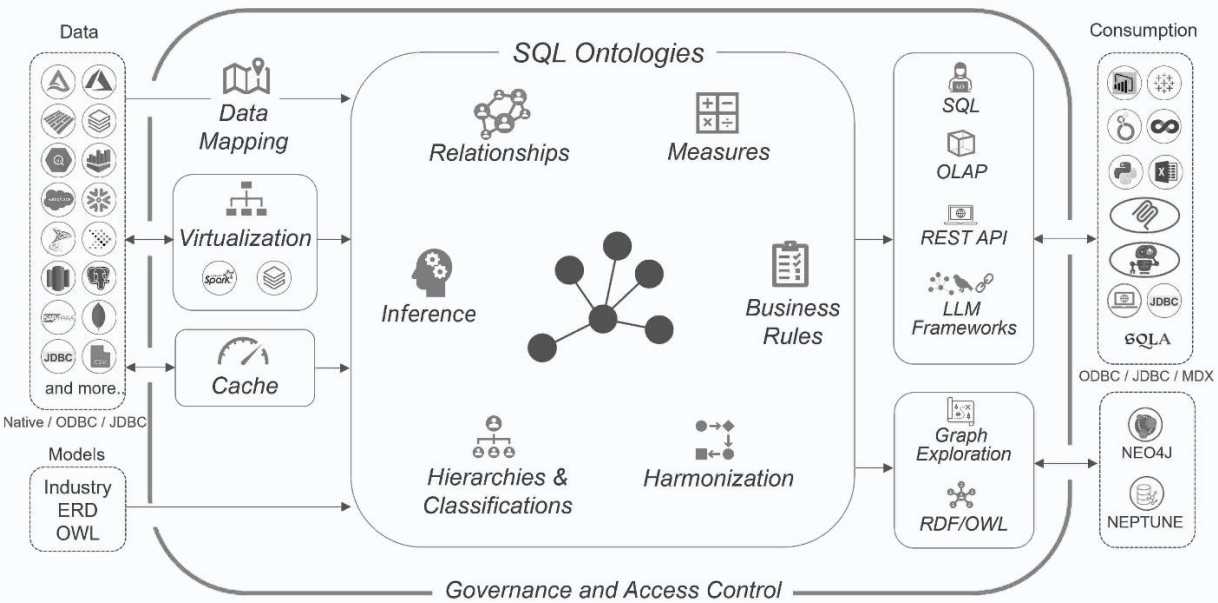
What is Timbr?

Timbr is an **SQL ontology-based semantic platform** that serves as **the enterprise semantic foundation for scalable, enterprise-grade LLM applications and autonomous agents**, enabling teams to integrate and consume complex data intuitively, collaborate seamlessly, and build faster insights. It transforms distributed, heterogeneous enterprise data into a governed, executable semantic layer that enables Large Language Models, BI tools, and applications to interact with data accurately, consistently, and at scale.

By modeling business concepts, relationships, hierarchies, and measures directly in standard SQL, Timbr provides the structured context required for reliable natural language-to-SQL (NL2SQL) generation, explainable AI responses, reusable metrics, and scalable data products, without ingesting, copying or transforming data.

Timbr SQL ontologies form a virtual semantic layer that connects directly to one or more underlying data sources. This layer acts as a logical abstraction over distributed systems, mapping physical tables, views, and fields into a unified model of business meaning. By decoupling business logic from physical storage, Timbr enables human and machine users to interact with complex multi-source data as a single semantic graph.

The platform includes a built-in virtualization and caching engine that dynamically executes optimized SQL across source systems. It is architecture-agnostic and installs on-premises, in the cloud, or in hybrid environments, providing the flexibility, scalability, and security required by modern data architectures.



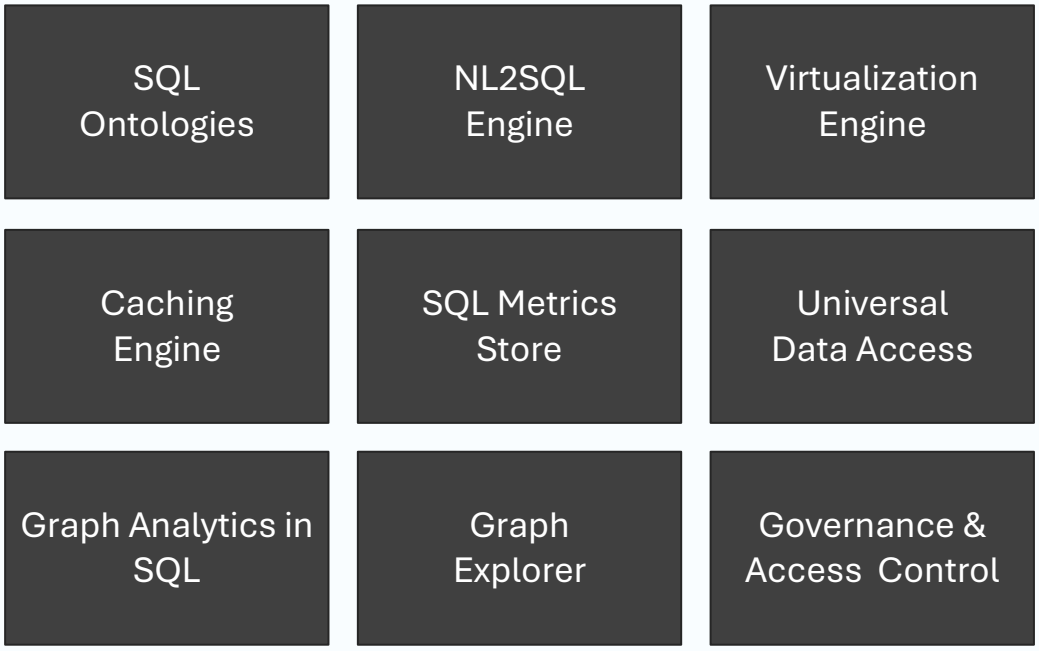
Why This Matters for Enterprise GenAI

Most enterprise LLM initiatives struggle with ambiguity, inconsistent metrics, and fragile prompt engineering when accessing structured data. Timbr eliminates this risk by grounding LLMs in a governed SQL ontology, providing explicit relationships, reusable measures, and executable semantic constraints.

The result is higher NL2SQL accuracy, reduced hallucinations, and fully traceable responses.

Core Capabilities Stack

The core capabilities stack is the foundation of the Timbr platform, built around the semantic layer to ensure seamless integration with on-premises, cloud, and hybrid data sources, alongside standardized measures, governed self-service, AI-ready connectivity, and broad consumption enablement through BI tools, natural language, and APIs.



Why Timbr

In the era of GenAI, semantics are infrastructure. Schemas describe storage. Ontologies describe meaning and relationships, enabling a shared understanding across teams and systems. They go beyond schemas by capturing business logic, hierarchies, and context, making data easier to explore, reuse, and govern.

Timbr's SQL ontologies bring this power into the SQL ecosystem, allowing teams to model data semantically while keeping it accessible in SQL. By replacing complex SQL JOINs with semantic relationships, Timbr significantly shortens and simplifies queries, making models easier to query, understand, and maintain.

As enterprises evolve toward distributed architectures, governed self-service, and AI-readiness, Timbr acts as **foundational infrastructure** for the modern data stack. It enables organizations to model meaning, standardize logic, and virtualize access across complex environments.

Timbr as Core Infrastructure for the Modern Data Stack

Ontology based Semantic Layer	Enterprise LLM Foundation	Data Virtualization	Metrics Store
Creates a unified, business-friendly view of data by mapping complex schemas to meaningful concepts, enabling self-service access and consistent governance.	Timbr LLM Foundation is the enterprise-grade backbone that empowers Large Language Models to interact seamlessly with structured, governed, and contextualized data.	Zero copy data integration from multiple sources. Timbr virtualizes disparate systems through a unified SQL ontology.	Centralize the definition and governance of business metrics and KPIs. Timbr enables teams to manage, reuse, and query metrics with full transparency.

Data Fabric	Data Mesh	Semantic Data Lake
Act as the semantic backbone for data fabric implementa-tions by connecting data sources, business logic, and governance across domains.	Empower domain teams to build interoperable, self-serve data products with governed ontologies and shared semantic contracts.	Transform data lakes into semantically enriched, queryable ecosystems, ready for analytics, AI, and natural language interaction.

How Timbr Works

The power of Timbr lies in how its core capabilities integrate to transform complex, distributed data into a unified, accessible, and governed semantic layer:

1. **Data Connectivity**
   Timbr connects directly to existing databases, data warehouses, data lakes, and APIs, with no need to move or replicate data.
2. **Semantic Modeling**
   Source datasets are defined as business concepts using a no-code interface or in standard SQL and organized into ontologies within the semantic layer, making data easier to understand and reuse.
3. **Query Federation**
   Timbr's virtualization engine dynamically routes and executes queries across relevant sources *only when multiple data sources are required*, abstracting away underlying complexity.
4. **Performance Optimization**
   A four-tier caching engine enables you to cache data selectively, from cost-efficient storage to ultra-fast in-memory layers, balancing query performance with infrastructure cost.
5. **Universal Access Layer**
   All BI tools, business applications, and AI systems consume data through a single semantic model, ensuring consistency in metrics, definitions, and governance across the organization.
6. **Governance and Access Control**
   Timbr provides built-in governance capabilities, including role-based access control, model versioning, data lineage, and audit trails, enabling secure, and compliant data usage across teams.

Key Outcomes

Timbr delivers consistent, governed, and high-performance access to data across the organization, accelerating insights while reducing complexity and infrastructure overhead.

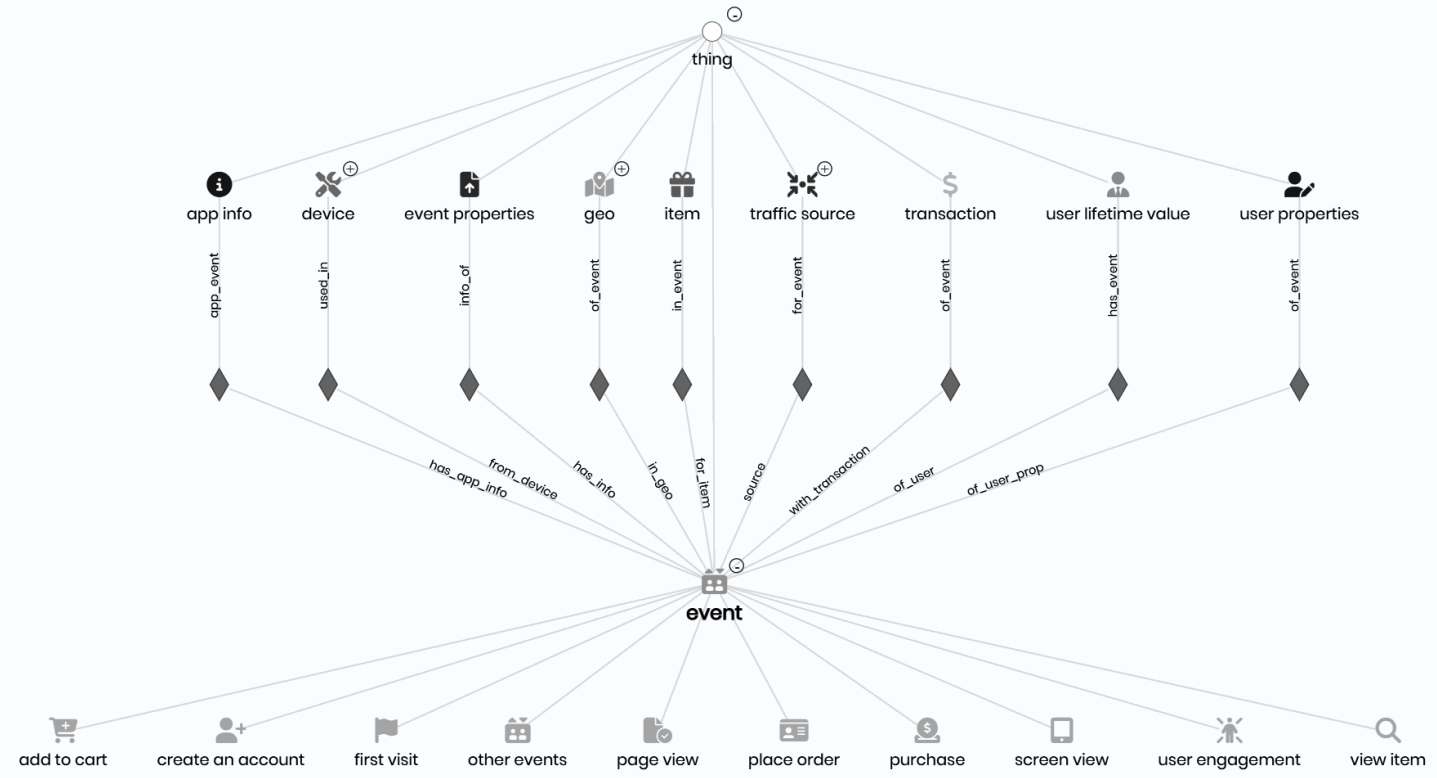
- **Single Source of Truth**
  Ensure consistent, governed business definitions across all teams and tools.
- **Accurate AI Integration**
  Empower precise LLM queries through semantic context and SQL-based relationships modeled in the ontology.
- **Reduced Data Movement**
  Eliminate costly ETL pipelines through virtualized access and smart caching.
- **Accelerated Time-to-Insight**
  Deliver data products and analytics faster through agile data engineering workflows.
- **Simplified Governance**
  Apply role-based access controls and data security policies directly at the semantic layer.
- **Enhanced Collaboration**
  Align business and technical users with a shared vocabulary and conceptual model.
- **Future-Proof Architecture**
  Easily onboard new data sources without changing downstream applications or dashboards.

Timbr's architecture empowers organizations to build and scale sophisticated semantic data models without disrupting existing infrastructure or embarking on expensive data movement or transformation.

SQL Ontologies: Capabilities

Timbr's SQL-native ontologies (W3C-stack equivalent+ in SQL), enable direct integration with enterprise data platforms and BI tools while maintaining graph semantics:

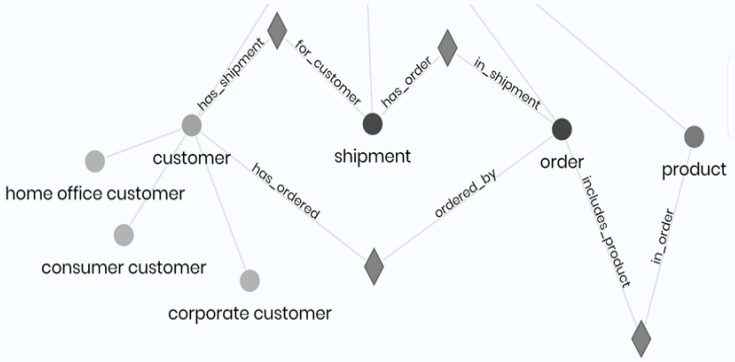
- **LLM/AI Readiness**
  Provide structure and context that enriches natural language-to-SQL conversion and optimizes LLM and autonomous agents' response accuracy.
- **Semantic Consistency Across Tools**
  Establish a single source of meaning across SQL, BI, and AI tools using reusable concepts and relationships.
- **Centralized Metrics & KPIs**
  Define, manage, and share trusted calculations and business definitions from one semantic layer.
- **Complex Data Modeling**
  Support sophisticated domains through inheritance, classifications, and cross-domain relationships.
- **Data Discovery & Reusability**
  Enable intuitive exploration and reuse through concept-driven navigation, not raw tables.
- **Decision Logic & Rule Modeling**
  Encode business rules and logic flows as semantic decision trees, queryable and transparent.
- **API Generation (OpenAPI/Swagger)**
  Generate self-documented APIs directly from the ontology model to enable semantic APIs and data-as-a-product delivery.
- **Governance & Access Control**
  Apply role-based access and semantic policies at the concept level.
- **Data Quality & Lineage**
  Model rules and relationships that validate, trace, and explain how data is produced.
- **Interoperability (RDF/OWL Export)**
  Export ontology models to RDF/OWL for integration with graph-native or linked data systems.



Semantic Relationships and SQL

What Are Semantic Relationships?

Semantic relationships define meaningful connections between data **concepts**, such as "Customer *has_ordered* Order" or "Order *in_shipment* Shipment" (which can also be defined in reverse). Unlike traditional database joins, **these relationships are expressed at the logical level**, reflecting how business entities relate in the real world. They form the backbone of Timbr's semantic layer, allowing users to work with data in terms they intuitively understand.



Modeling Relationships

Semantic relationships are modeled in the ontology to link between the concepts representing (and mapped to) data tables and columns. Timbr supports multiple types of relationships within the semantic model, including **one-to-many**, **many-to-many**, and **transitive (recursive)** relationships. These are defined once in SQL and automatically reused across queries, enabling advanced logic like **rollups**, **drilldowns**, and **recursive paths** without manually defining JOIN chains.

How They Work

When users query data through Timbr, semantic relationships eliminate the need to manually write complex SQL JOINs. Instead, Timbr interprets the defined relationships and **automatically generates an optimized, high-performance query in standard SQL, which is pushed down to the database in its native dialect.** Queries become simple and intuitive expressions, requiring no knowledge of underlying table structures or join conditions, so data teams accelerate delivery of data products.

Key Outcomes

- **Shorter, clearer SQL:** Replace verbose joins with clean semantic paths.
- **AI-readiness:** Relationships reflect true business logic, delivering accurate NL2SQL and LLM-based insights.
- **Reusable logic:** Define relationships once and use them everywhere.
- **Faster onboarding:** New users query data using familiar business concepts.
- **Improved governance:** Centralized control of how data entities relate.
- **Advanced analysis:** Rollups, drilldowns, and recursive navigation made simple.
- **Cross-source integration:** Relate concepts across systems without physically merging data.

SQL Query with JOINS

```sql
SELECT `order`.`order_id`, `order_date`,
  `home_office_customer`.`customer_name`,
  `first_class_shipment`.`shipping_date`,
  `fitness_product`.`product_name`
FROM `scdata`.`datascupplychaindataset_order` as `order`
LEFT JOIN (
  SELECT product_id, product_name
  FROM `supply_chain_item`.`product`
  WHERE `department` = 'fitness') as `fitness_product`
ON `order`.`product_id` = `fitness_product`.`product_id`
LEFT JOIN (
  SELECT `order_id`, `shipping_date`
  FROM `scdata`.`final_mayshipmentfinal_mayshipment`
  WHERE `shipping_mode` = 'First Class') as `first_class_shipment`
ON `order`.`order_id` = `first_class_shipment`.`order_id`
LEFT JOIN (
  SELECT `customer_id`,
    concat(`customer_first_name`, ' ', `customer_last_name`) as `customer_name`
  FROM `scdata`.`datascupplychainlatest_customer`
  WHERE `customer_segment` = 'Home Office') as `home_office_customer`
ON `order`.`customer_id` = `home_office_customer`.`customer_id`
WHERE `product_price` * `order_item_quantity` - `order_item_discount` > 500
```

SQL Query with Relationships

```sql
SELECT order_id, order_date,
  `ordered_by[home_office_customer]`.customer_name,
  `in_shipment[first_class_shipment]`.shipping_date,
  `includes_product[fitness_product]`.product_name
FROM dtimbr.`order`
WHERE revenue > 500
```

Core Capability

SQL Ontologies Manager

The Ontology Manager is the foundation of Timbr's semantic modeling capabilities, empowering data teams to build and govern rich, queryable ontologies that reflect how their organization understands and uses data.

The ontology modeler enables users to design rich semantic representations of their data by defining **concepts** (business entities), their **relationships** (semantic joins), and structured **hierarchies** that reflect real-world classifications. Users can organize concepts into parent-child structures to support **inheritance** and **concept specialization**, allowing complex domains to be modeled with clarity and reuse. The modeler also supports defining **measures**, **calculated properties**, and embedding **business logic** directly into the ontology through rule-based filters or computed fields. In addition, it provides tools to **clean, filter, and transform data virtually**, ensuring semantic accuracy without altering the underlying data sources. This approach allows users to create a unified, governed, and highly consumable semantic layer.

Flexible Ontology Creation

Timbr supports both **bottom-up** and **top-down** ontology generation:

- **Bottom-Up**: Enables teams to model their data based on existing schemas. With Timbr's visual, no-code interface, users can browse their databases and generate AI-assisted **automatic recommendations for concepts, relationships and measures**, accelerating modeling and reducing human error.
- **Top-down creation** offers multiple advanced options:
  - **Manual DDL-based modeling**, using standard SQL commands to define concepts and relationships.
  - **LLM-assisted generation**, where large language models translate ERD diagrams or documentation into DDL scripts.
  - **OWL ontology translation**, converting RDF/OWL-based ontologies into Timbr-compatible SQL structures.
  - **Catalog conversion**, transforming metadata catalogs into semantic models automatically.

Timbr's data modeling flexibility allow organizations to model knowledge at their own pace and level of complexity, whether they're starting from scratch or enriching existing architectures.

Modeling within Databricks Notebooks

A standout feature of Timbr is its **native integration with Databricks notebooks**, allowing users to define and manage semantic models directly within their existing data pipelines. This integration offers:

- **Declarative Semantic Modeling**: Define business concepts, relationships, and rules as part of the data pipeline, ensuring alignment between data processing and business logic.
- **Enhanced Data Quality**: Implement data cleaning and validation rules that are applied dynamically as data is accessed, promoting consistent and accurate analytics.
- **Seamless Governance**: Integrate with Databricks Unity Catalog to centralize management of data definitions, relationships, and quality rules.

By embedding semantic modeling capabilities within Databricks, Timbr streamlines workflows and fosters a more cohesive data environment.

Comprehensive Ontology Management

Beyond creation, the Ontology Manager provides robust tools for managing and exploring ontologies:

- **Graphical Exploration**: Visualize and navigate the semantic model to understand data relationships and hierarchies.
- **Relationship Customizations**: Support for hierarchies, inheritance, and logic rules, all applied at **query time** , ensuring data accuracy without altering underlying sources.
- **Built-in data filtering and cleaning**: Apply filters and data cleaning operations during query execution, ensuring real-time data standardization and refinement.
- **View Management**: Create and manage virtualized database views, facilitating reusable and consistent data representations across the organization.

Core Capability

Enterprise LLM Foundation

Enterprise LLM Foundation is the enterprise-grade backbone that empowers Large Language Models and autonomous agents to interact seamlessly with structured, contextualized, and governed data. LLM-F leverages semantic modeling, relationships, SQL measures, and virtualization to power LLM workflows via GraphRAG, LangChain and LangGraph, while supporting MCP-based orchestration and multi-agent systems. The result is a scalable, explainable foundation for AI-driven analytics and intelligent automation across the enterprise.

LangChain SDK

The Timbr LangChain SDK enables developers to integrate natural language interfaces with Timbr's semantic layer. It utilizes LLMs to interpret user queries, generate Timbr SQL, and fetch query results from the database. The SDK supports multi-LLM integration with the popular models, Timbr's native LLM and custom LLMs as well.

Key Features:

- **Multi-LLM Support:** Integrates with LLMs, providing flexibility in choosing the appropriate model for specific use cases.
- **SQL Generation:** Generates semantic SQL queries enriched with ontology context.
- **Streamlined Querying:** Combines natural language inputs with Timbr using simple methods like run_llm_query.

LangGraph SDK

The Timbr LangGraph SDK enables developers to build multi-step natural language workflows that interact with Timbr's semantic layer, leveraging its ontology to generate context-aware SQL queries. It leverages Timbr's robust ontology capabilities, integrating with Timbr data models and leveraging semantic relationships and annotations,.

Key Features:

- **Concept Identification:** Identifies the relevant concept and schema based on the user's query.
- **SQL Generation:** Generates SQL from a natural language prompt, returning the generated SQL along with the schema and concept.

Key Benefit:
Purpose-Built for the NL2SQL Paradigm

The rise of large language models as analytical interfaces introduces a critical shift: translating natural language into precise, context-aware SQL. Timbr is uniquely equipped to meet this challenge.

Its SQL-native knowledge graph models business concepts, relationships, and measures directly in SQL, providing the structured, interpretable context LLMs need to generate accurate queries, reducing hallucination and enforcing metric governance.

Unlike traditional semantic layers and SPARQL-based knowledge graphs, Timbr offers the operational integration and expressiveness essential for reliable NL2SQL generation. Moreover, LLMs are far better trained to generate SQL than SPARQL due to the volume of SQL examples in their training data, making Timbr's architecture inherently more compatible with LLM reasoning.

GraphRAG SDK

The Timbr GraphRAG SDK equips developers with a powerful toolkit to build Retrieval-Augmented Generation (RAG) applications enhanced by Timbr's SQL-based knowledge graph. By grounding large language models in Timbr's semantically enriched ontology, the SDK delivers precise, context-aware answers from structured enterprise data. It leverages the ontology's logical relationships, metadata, and reasoning capabilities to ensure the LLM retrieves the most relevant knowledge and generates grounded, explainable outputs.

Key Features:

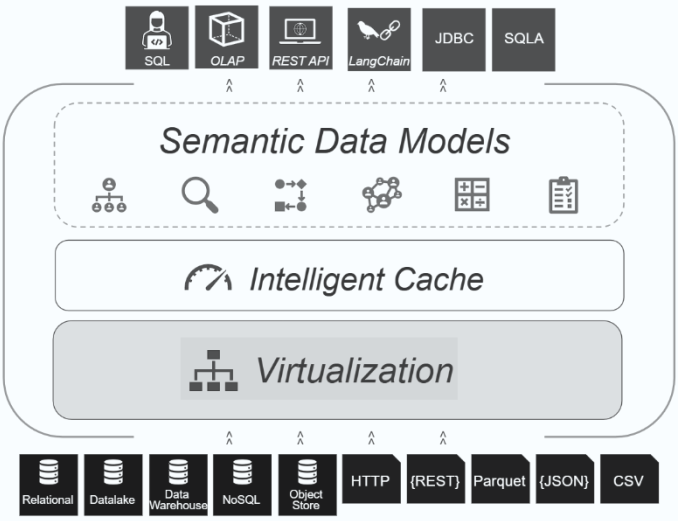
- **Ontology-Grounded Retrieval:** Enhances RAG workflows by retrieving context based on Timbr concepts, hierarchies, and transitive relationships.
- **SQL-Based Context Injection**: Automatically generates SQL queries to extract precise context from Timbr-modeled data sources.
- **Semantic Chunking & Re-ranking**: Segments and prioritizes retrieved content using semantic proximity, improving grounding accuracy.
- **Traceable Outputs**: Supports provenance by linking answers to their originating concepts and SQL queries.

Core Capability

Virtualization Engine

Timbr's Virtualization Engine is a cornerstone of Timbr for **enabling federated queries across distributed data without requiring physical movement or duplication.**
By abstracting the complexities of underlying sources, it presents a unified semantic graph that simplifies data consumption and analysis.

Powered by a distributed Spark engine, the Virtualization Engine scales with data, pushing down optimized SQL logic while maintaining high performance across large and complex workloads.



When an organization already has a virtualization layer in place such as Databricks SQL, Presto, or Trino, Timbr can integrate with that engine directly, extending its semantic capabilities without replacing existing infrastructure.

Core Capability

Caching Engine

Timbr's Caching Engine is designed to **enhance performance and efficiency by materializing data strategically**, reducing the need for repetitive computations and accelerating query responses.

Multi-Tiered Caching

Local Database: Stores cached data within the primary database used in the data model, ensuring proximity and quick access.

Data Lake: Utilizes object storage solutions like S3, Azure Data Lake Storage, or Google Cloud Storage for cost-effective caching of large datasets.

SSD Storage: Allocates solid-state drives within the Timbr cluster for intermediate performance needs, balancing speed and cost.

In-Memory Storage: Employs an in-memory database for the fastest access, ideal for real-time dashboards and high-frequency queries.

Key Features:

- **Multi-Tiered Caching:** Timbr offers four distinct caching tiers to accommodate cost-performance and storage requirements.
- **Flexible Materialization:** Timbr allows for the materialization of both mappings and ontology views. Users can define caching strategies using SQL commands, specifying schema names, partition properties, and even scheduling parameters for automated refreshes.
- **Partitioning and Splitting:** Enhance cache management by partitioning data based on properties like timestamps or categories. Further, split partitions (e.g., daily, monthly) to optimize query performance and cache refresh efficiency.
- **Integration with Object Storage:** Timbr supports caching directly into object storage systems, facilitating seamless integration with existing data lake architectures and enabling scalable storage solutions.
- **Automated and Manual Refresh:** Caches can be refreshed automatically at defined intervals or triggered manually. This flexibility ensures that cached data remains up to date, aligning with the latest data changes and business requirements.
- **Promotion and Demotion of Caches:** Depending on workload demands, cached data can be promoted to higher-performance tiers or demoted to more cost-effective storage options, providing dynamic resource management.

By incorporating these caching strategies, Timbr significantly reduces query execution times, optimizes resource utilization, and provides a robust foundation for scalable and responsive data operations across diverse environments.

Core Capability

SQL Metrics Store

Timbr's SQL Metrics Store enables users to define reusable, governed measures, including metrics, KPIs, and complex calculations, directly in the semantic model using standard SQL. It centralizes business logic, ensures consistency across reports and tools, and accelerates analytical project development without locking users into proprietary formats.

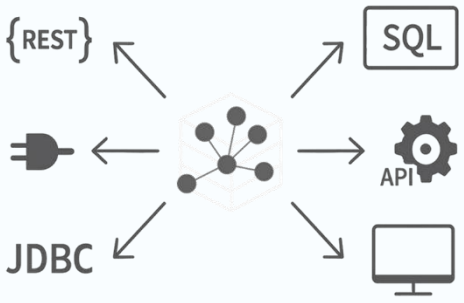
Key Features:

- **Centralized Metrics Definitions:** Measures are defined using SQL expressions embedded in the semantic model. This abstraction eliminates the complexity of joins and underlying schemas, allowing clear and maintainable metrics definitions.
- **Inheritance and Reuse:** Timbr supports hierarchical measure definitions, enabling new metrics to inherit from and extend existing ones. This promotes consistency and reduces redundancy across projects, departments, and analytics workflows.
- **Auto-Discovery and Suggestions:** The platform analyzes existing semantic models to suggest relevant measures based on data patterns and relationships. This accelerates the creation of business-relevant KPIs and reduces manual effort.
- **SQL-Native Access:** All measures are fully accessible through standard SQL and are compatible with any SQL-based tool, including BI platforms, notebooks, and data applications.
- **DAX-to-SQL Translation:** Timbr automatically translates DAX measures into equivalent SQL, simplifying the migration of Power BI models and preserving business logic while improving performance and portability.
- **Federated Metrics Computation:** When measures span multiple data sources, Timbr's virtualization engine dynamically routes and executes the necessary queries. This capability enables federated metrics computation without requiring data movement or duplication, and is only activated when multiple sources are involved.

Core Capability

Universal Data Access

Timbr provides a unified, governed access point to semantically modeled data, enabling a wide range of consumption methods tailored to the needs of business users, analysts, data scientists, and developers.



Regardless of the underlying data architecture, Timbr ensures that the same governed, semantically enriched view of data is accessible to every persona and tool in the organization. This reduces duplication of logic, improves collaboration, and enables organizations to operationalize data with greater confidence and speed.

SQL Connectivity Across Tools and Platforms

At the foundation of Timbr's accessibility is its native support for standard SQL, allowing users to query complex semantic models without needing to understand the physical data structure. This eliminates the need to write complex joins or recreate logic in downstream tools, reducing errors and accelerating insight generation. Queries are pushed down to source systems for optimal performance, enabling real-time access to harmonized data across multiple platforms, including Databricks, Snowflake, Synapse, BigQuery, and others.

Seamless Integration with BI Tools: ODBC, JDBC, and MDX Connectivity

To ensure interoperability across the analytics ecosystem, Timbr supports industry-standard interfaces:

- **ODBC and JDBC** allow integration with virtually any BI, data integration, or reporting tool.
- **MDX** enables multidimensional querying and supports hierarchical exploration in OLAP-style tools, including Excel pivot tables and other cube-aware environments.

These options make the semantic layer broadly consumable without requiring proprietary connectors or changes to existing workflows.

Analysts can connect directly to Timbr from tools such as Power BI, Tableau, Looker, and Excel, accessing governed metrics and semantic definitions embedded in the model. This ensures consistent reporting logic across tools, departments, and data sources, reducing errors and duplication while accelerating dashboard creation.

Data Science and Engineering Access

Data scientists and engineers can access semantically modeled data from their preferred environments, including **Python, R, Scala, and Java**. Timbr provides compatibility with libraries such as **SQLAlchemy via its Python SDK**, making it easy to integrate semantic data into notebooks and machine learning pipelines. This reduces time spent on data wrangling and improves the quality and interpretability of AI models by ensuring consistent definitions and relationships.

MCP-Ready Programmatic Access via REST API - Self Documented

Timbr also offers a comprehensive **RESTful API** for accessing and interacting with the semantic model and its data. The MCP ready API supports advanced filtering, metadata queries, and dynamic data retrieval, making it easy to embed semantic insights into operational applications and workflows. The implemented APIs are self-documented using **Swagger/OpenAPI**, allowing developers to explore and test endpoints directly, accelerating integration and reducing the need for custom documentation.

Core Capability

Governance and Access Control

Timbr's Governance and Access Control is a robust, enterprise-grade solution designed to provide enterprise-grade oversight of data access, usage, and model integrity. It enforces fine-grained permissions across data and platform operations, with consistent policy enforcement across SQL, BI, and API access.

Built-in capabilities like role-based access control, model versioning, data lineage, and audit trails ensure secure, transparent, and compliant use of the semantic layer.

Integration with Enterprise Identity Systems

To support seamless user management, Timbr integrates with enterprise identity providers:

- **Single Sign-On (SSO):** Leverage SSO mechanisms for streamlined authentication.
- **Azure Active Directory (AD) and AD Roles**: Integrate with Azure AD for role-based access control, aligning Timbr's permissions with existing organizational structures.

Key Features

- **User and Role Management**
  Centrally manage users, assign custom roles, and group permissions by project, domain, or business function.
- **Granular Access Control**
  Define and enforce permissions at multiple levels, from platform features and semantic objects to concepts, datasources, mappings, and views.
- **Data Table-Level Permissions**
  Restrict access to specific tables and views within the semantic model, ensuring users only interact with approved data assets.
- **Row-Level Security**
  Enforce fine-grained access by controlling visibility at the individual record level, tailored to user identity, roles, or business context.
- **Roles & Policies-Based Governance**
  Apply reusable access rules across users and teams to support consistent governance practices organization-wide.
- **Enterprise Identity Integration**
  Seamlessly align with corporate identity providers through SSO and directory-based role mapping to streamline user onboarding and policy enforcement.

Model Versioning

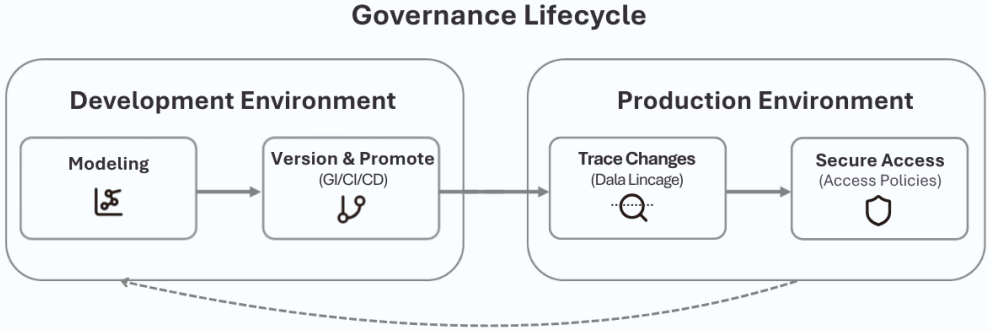
Semantic model versioning capabilities allow teams to maintain snapshots of the ontology and its components, enabling safe iteration, rollback, and reproducibility.

Data Lineage

Timbr provides full visibility into how data flows through the semantic layer, from raw sources to final modeled outputs. Data lineage captures the dependencies between source tables, transformations, and conceptual views, allowing users to trace how definitions are built, and which datasets or metrics are affected by changes.

Audit Trails and Activity Logging

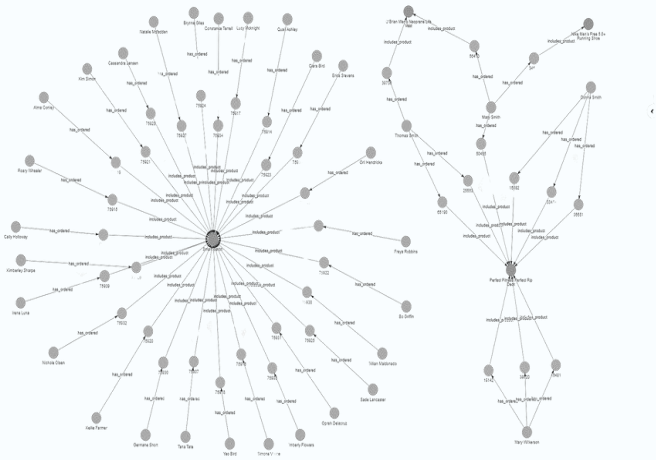
All user interactions and administrative actions are logged, offering a complete audit trail of platform activity. This includes changes to models, permission assignments, query executions, and API access. Organizations can review, monitor, and report on user behavior to support internal governance processes, security reviews, and external compliance requirements.



Capability

Graph Explorer

Timbr's Graph Explorer is an intuitive interface that allows users to visually navigate and analyze data relationships within the semantic model. By representing data as interconnected nodes and edges, it facilitates a deeper understanding of complex data structures without the need for extensive technical knowledge.



Visualizing Data Relationships

Graph Explorer presents data entities and their interconnections in a graphical format, enabling users to:

- Identify and comprehend relationships between different data concepts.
- Discover patterns and dependencies that may not be evident through traditional tabular views.
- Gain insights into data lineage and the flow of information across systems.

Enhancing Data Exploration

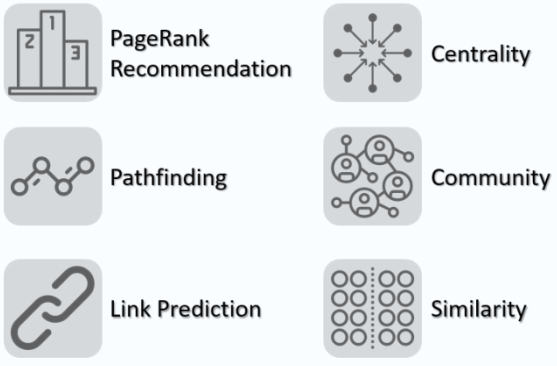
By providing a visual representation of the semantic model, Graph Explorer aids in:

- Facilitating data discovery and comprehension for both technical and non-technical users.
- Supporting data quality assessments by highlighting anomalies or unexpected relationships.
- Accelerating the onboarding process for new users by offering an accessible overview of the data landscape.

Capability

Graph Analytics in SQL

Timbr's Graph Analytics module empowers organizations to perform advanced graph computations directly on their existing relational data, eliminating the need for specialized graph databases or complex data transformations.



Key Features

- **Direct Execution on Relational Data**: Run graph algorithms on data stored in platforms like Azure, BigQuery, Athena, MySQL, Oracle, SAP, and other SQL-compatible databases without the need to convert data into a graph format.
- **Integration with BI Tools**: Access and execute graph analytics seamlessly from popular business intelligence tools, enabling analysts to incorporate complex relationship insights into their reports and dashboards.
- **Support for Advanced Frameworks**: Materialize graph computations to advanced frameworks such as NVIDIA RAPIDS cuGraph, Spark GraphFrames, and Python's NetworkX, allowing for scalable and high-performance graph processing.
- **Automated Ontology Enrichment**: Automatically enhance the semantic model with insights derived from graph algorithms, ensuring that the ontology evolves with the data's underlying relationships.

Timbr offers a curated library of graph algorithms tailored for business applications, which may be expanded per customer requests.

Build Your Enterprise LLM Foundation with Timbr.
Get started with a free POC*

START YOUR FREE POC

* Subject to Terms – See POC registration page

Ver2.4