

METADATALOOP PLATFORM

User Guide

Schema intelligence, lineage mapping, quality monitoring,
and developer tooling — unified for data engineering teams.

Generated April 25, 2026
Version 2.4.1

Contents

1. Core Tools — Schema Intelligence

- 1.1 ■ Schema Explorer
- 1.2 ■ Schema Insights
- 1.3 ■ Data Dictionary
- 1.4 ■ SchemaSpy View

2. Core Tools — Modeling & Comparison

- 2.1 ■ Data Lineage
- 2.2 ■ Data Quality
- 2.3 ■ Dimensional Model
- 2.4 ■■ Schema Diff

3. Core Tools — Developer Utilities

- 3.1 ■ SQL Formatter
- 3.2 ■ Schema → Markdown
- 3.3 ■ Mermaid / DBML Converter
- 3.4 ■■ SQL Converter (T-SQL → Databricks)

4. Core Tools — Snowflake & Migrations (NEW)

- 4.1 🌸■ Snowflake Studio
- 4.2 ■ Migration Generator
- 4.3 ■ Mock Data Generator
- 4.4 ■ Schema Inferer

5. Premium Tools

- 5.1 ■ Enterprise Data Discovery
- 5.2 ■ KPI Metrics Tracker
- 5.3 ■ AI Analytics & Reporting
- 5.4 ■ AI Documentation

Introduction

MetaDataLoop is a unified workspace for data engineers — combining live database exploration, lineage and quality tooling, and a growing catalogue of dialect-aware developer utilities. This guide walks through every tool on the platform, when to reach for it, and exactly which buttons to click.

Conventions used in this guide

Bold UI labels match buttons, tabs, and field names exactly as they appear in the application. `Monospaced text` indicates code, file names, or example identifiers. Tips appear in a tinted callout below the steps.

Getting started

Sign in, then open the home page. Tools are organised under **Core Tools (Free)** — available to everyone — and **Premium Tools**, which unlock with a paid plan. All free tools work offline on pasted SQL or sample data; the live-DB tools require outbound connectivity to your database.

1. Core Tools — Schema Intelligence

1.1 ■ Schema Explorer

Interactive ER browser for live databases — Postgres, MySQL, SQL Server, Snowflake. Click any table to see columns, types, and relationships.

When to use it

You inherited an unfamiliar database and need to understand its shape quickly without writing SQL.

How to use it

1. Open **Core Tools** → **Schema Explorer**.
2. Pick a saved connection or add a new one (Host, Port, Database, Username, Password).
3. The left pane lists schemas and tables; click a table to inspect columns, PKs, FKs and sample relationships.

1.2 ■ Schema Insights

Static metadata intelligence extracted directly from a SQL file — no DB connection required.

When to use it

You only have a SQL dump or DDL script and want a quick structural overview.

How to use it

1. Upload a .sql file containing CREATE TABLE statements.
2. Schema Insights extracts table counts, column profiles, naming patterns, and relationship hints.

1.3 ■ Data Dictionary

Generate business-friendly dictionaries with descriptions for every table and column.

When to use it

Stakeholders need to understand what each column means without reading the DDL.

How to use it

1. Connect to a live database *or* upload a SQL file.
2. Optionally provide column descriptions; the tool fills gaps based on naming heuristics.
3. Export the dictionary as a downloadable artifact.

1.4 ■ SchemaSpy View

Visualize ER diagrams in a SchemaSpy-style layout — without the Java dependency.

When to use it

You want a familiar SchemaSpy-style report but inside the browser.

How to use it

1. Provide a connection or upload a schema file.
2. Browse table-level diagrams, FK relationships, and orphan-table reports.

2. Core Tools — Modeling & Comparison

2.1 ■ Data Lineage

Visualize column-level lineage and SQL dependencies across your entire pipeline.

When to use it

You need to know what feeds into a column or what depends on it before making a change.

How to use it

1. Upload SQL scripts (CREATE TABLE, CREATE VIEW, INSERT...SELECT, etc.).
2. Lineage parses the references and renders an interactive graph.
3. Click any node to highlight upstream and downstream dependencies.

2.2 ■ Data Quality

Run automated checks, detect anomalies, and maintain data trust across all datasets.

When to use it

You want to monitor freshness, null rates, distinct counts, and rule violations.

How to use it

1. Open **Core Tools** → **Data Quality**.
2. Connect a live DB or load schema metadata.
3. Pick or define checks; results appear with pass/fail status and drill-down details.

2.3 ■ Dimensional Model

Build star schemas with automatic PK/FK detection and fact/dimension tagging.

When to use it

You're designing or auditing a warehouse model and want to spot fact-vs-dim candidates.

How to use it

1. Paste or upload your CREATE TABLE statements.
2. The tool analyses cardinality and key relationships, suggesting fact and dimension labels.
3. Tweak the suggestions and export the resulting model.

2.4 ■■ Schema Diff

Highlight structural differences between two SQL schemas, line-by-line and column-by-column.

When to use it

Comparing dev vs prod, or pre/post a migration — quickly identifying what changed.

How to use it

1. Upload two SQL files.
2. View a unified diff with added, removed, and changed lines.
3. Download the diff for code review or attaching to a ticket.

3. Core Tools — Developer Utilities

3.1 ■ SQL Formatter

Format any SQL with consistent indentation, keyword casing, and spacing.

When to use it

Cleaning up gnarly SQL before code review or pasting into documentation.

How to use it

1. Paste SQL into the input pane.
2. Choose keyword case (UPPER / lower / Capitalize), indent width, and whether to strip comments.
3. Click **Format**; copy or download the result.

3.2 ■ Schema → Markdown

Turn CREATE TABLE statements into clean Markdown docs for GitHub, Notion, or wikis.

When to use it

You want to commit human-readable schema docs alongside your code.

How to use it

1. Paste CREATE TABLE statements (each ending with ;).
2. Click **Generate Markdown**.
3. Copy the rendered Markdown or download schema.md.

3.3 ■ Mermaid / DBML Converter

Convert SQL schemas to Mermaid erDiagram or DBML markup for dbdiagram.io.

When to use it

Embedding diagrams in Markdown (Mermaid) or generating shareable diagrams (DBML).

How to use it

1. Paste CREATE TABLE statements.
2. Toggle between **Mermaid** and **DBML**.
3. Copy the output or download .mmd / .dbml.

3.4 ■■ SQL Converter (T-SQL → Databricks)

Convert T-SQL DDL into clean Databricks-compatible SQL with zero manual effort.

When to use it

Migrating SQL Server objects to Databricks / Spark SQL.

How to use it

1. Upload or paste T-SQL DDL.
2. Run the conversion and review the output.
3. Copy or download the Databricks-ready SQL.

4. Core Tools — Snowflake & Migrations (NEW)

4.1 ❄️ ■ Snowflake Studio

Save and manage your Snowflake accounts, browse warehouses and databases, and run ad-hoc SQL — all in one place.

When to use it

You work with multiple Snowflake accounts and want a single UI for connection management and quick exploration.

How to use it

1. Open **Core Tools** → **Snowflake Studio**.
2. Add a new account (Account identifier, Username, Password / Key, Warehouse, Role).
3. Pick the active account and browse databases / schemas / tables.
4. Run ad-hoc SQL inside the integrated query panel.

4.2 ■ Migration Generator

Diff two SQL schemas and emit ready-to-run **ALTER / CREATE / DROP** migration scripts for Postgres, MySQL, Snowflake, or SQL Server.

When to use it

You changed your CREATE TABLE in code and need the corresponding production migration.

How to use it

1. Paste the **Current Schema** on the left.
2. Paste the **Target Schema** on the right.
3. Pick the dialect — the script uses dialect-specific syntax (e.g. MySQL MODIFY COLUMN, Snowflake SET DATA TYPE).
4. Click **Generate Migration**; review the stats chips and copy or download the script.

Tip. Detected differences include new tables, removed tables, added/removed columns, and column type or NULL changes. Drop and modify operations are destructive — always review before running in production.

4.3 ■ Mock Data Generator

Generate realistic test rows from any CREATE TABLE — names, emails, dates, amounts — exported as CSV, JSON, or SQL INSERT.

When to use it

You need seed data for dev / staging / unit tests without exposing real customer data.

How to use it

1. Paste your CREATE TABLE statements.
2. Choose rows per table (1 – 5000), output format, and (for SQL) the dialect.
3. Optional: enter a **Seed** for reproducible runs.
4. Click **Generate**; Copy / Download buttons activate when output is ready.

Tip. Column-name heuristics override SQL types — a column named email will get fake emails even if typed VARCHAR. Recognised names include email, first_name, last_name, phone, address, city, status, price, qty, created_at, is_active, and more.

4.4 ■ Schema Inferrer

Paste a CSV (with header) or JSON sample — get a typed CREATE TABLE for Postgres, MySQL, Snowflake, or SQL Server.

When to use it

You have a sample data file and want a starting-point schema without typing it by hand.

How to use it

1. Paste your CSV or JSON sample (auto-detected by default).
2. Set the table name and dialect.
3. Click **Infer Schema**; review the column types in the stats bar.
4. Copy or download the generated .sql.

Tip. Detected types include INT/BIGINT, DECIMAL, BOOLEAN, DATE, TIMESTAMP, UUID, and VARCHAR(N) where N rounds up to 16/32/64/128/255/512/1024 (over 1024 → TEXT). Columns with any null value are marked nullable.

5. Premium Tools

5.1 ■ Enterprise Data Discovery

Locate any value across your entire schema ecosystem instantly using high-performance indexed search.

When to use it

Investigating which tables and columns store a particular value (a customer ID, an email, a SKU).

How to use it

1. Connect the databases you want to index.
2. Run an initial index build.
3. Search by value or by column-name pattern; results show table, column, and match count.

5.2 ■ KPI Metrics Tracker

Comprehensive metrics dashboard with tabbed interface and data persistence.

When to use it

Centralising recurring KPI reports across multiple data domains.

How to use it

1. Define metrics (name, query, refresh cadence).
2. Group them by tab.
3. Open the dashboard for current values, trends, and history.

5.3 ■ AI Analytics & Reporting

Generate insights, dashboards, and reports from your data using AI-powered natural-language queries.

When to use it

Non-SQL users want to explore data conversationally.

How to use it

1. Connect a data source (file upload, Postgres, MySQL, Snowflake, BigQuery, SQL Server, Redshift).
2. Ask a question in plain English (e.g. "Show me monthly revenue by region").
3. Review the auto-generated chart, insight summary, and SQL.

5.4 ■ AI Documentation

Connect to a saved Snowflake account and auto-generate enterprise-grade schema documentation (DOCX + PDF) with GPT-4o.

When to use it

You need polished, branded schema documentation for stakeholders or auditors.

How to use it

1. Pick a saved Snowflake account.
2. Choose the database and tables to include.
3. Click **Generate**; download the resulting DOCX and PDF.

Support & Contact

MetaDataLoop is built and maintained by AppLabsTech Solutions Inc. For questions, feedback, or demo requests, get in touch using any of the channels below.

Email

service@metadataloop.com

Phone

+1 (903) 443-1188

Address

AppLabsTech Solutions Inc.
6671 Southwest Fwy, Suite 700
Houston, Texas 77074, United States

LinkedIn

linkedin.com/company/metadataloop