SQL Identity Resolution - One-Page App Summary

Repo-based summary only (no external claims)

What It Is

SQL Identity Resolution (IDR) is a warehouse-native identity graph engine that links records across sources using deterministic SQL rules.
It can apply optional fuzzy matching refinement and produces resolved clusters plus golden profiles.

Who It Is For

Primary persona (inferred): data/analytics engineers and developers running identity resolution in SQL warehouses.
Explicit persona statement: Not found in repo.

What It Does

- Runs a unified pipeline (FULL/INCR) via the idr CLI on DuckDB, BigQuery, Snowflake, and Databricks.
- Builds deterministic identity edges from identifiers and resolves clusters with label propagation.
- Optionally applies fuzzy matching; strict mode disables fuzzy matching for deterministic-only runs.
- Supports dry-run previews, metadata init, and YAML config validate/generate/apply workflows.
- Exposes FastAPI endpoints for setup, dashboard metrics, explorer search, runs, schema, and health.
- Provides a React web UI with Setup Wizard, Dashboard, Explorer, Runs, Data Model, and Settings pages.

How It Works (Repo-Evidence Architecture)

- Metadata/rules live in idr_meta, transient processing in idr_work, and outputs in idr_out.
- IDRRunner orchestrates stages: Preflight -> Extraction -> Graph -> Output.
- Graph stage builds edges, runs label propagation, then optional fuzzy matching refinement.
- Output stage materializes current membership/golden profile tables and records run history.
- Interfaces: CLI and MCP use core engine; UI calls FastAPI routers that use platform adapters.

How To Run (Minimal)

1) pip install "sql-identity-resolution[duckdb]"
2) idr quickstart
3) Optional: idr serve, then open http://localhost:8000

Evidence Used

Readme.md, docs/01-overview.md, docs/18-architecture.md, docs/03-quickstart.md, idr_core/runner.py, idr_core/cli.py, idr_api/main.py, idr_ui/src/App.tsx