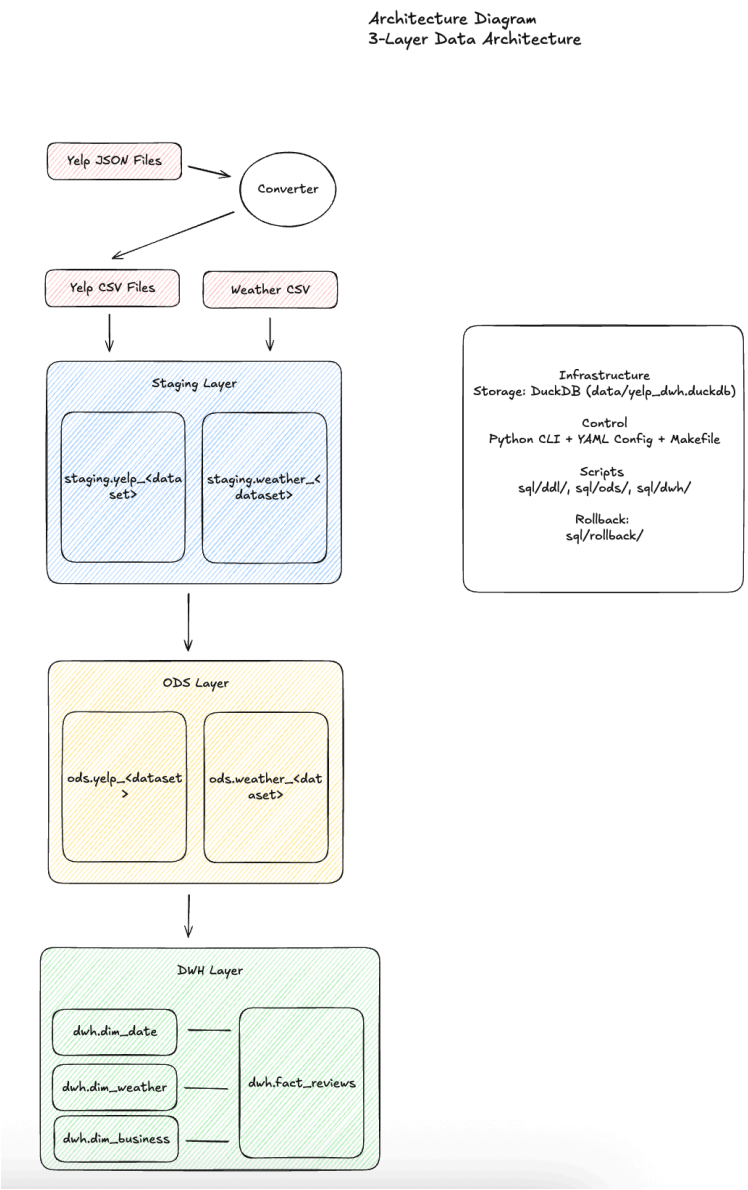
Yelp DWH

By Muchamad Fajar Alif

Architecture Diagram



3-Layer Data Architecture
Raw Data → Staging → ODS → DWH

Data Flow

1. Input: Yelp JSON files + Weather CSV files
2. Conversion: JSON → CSV (src/converter/)
3. Staging: Raw data loaded as-is (staging.* tables)
4. ODS: Cleaned & normalized (ods.* tables)
5. DWH: Star schema for analytics (dwh.* tables)

Key Components

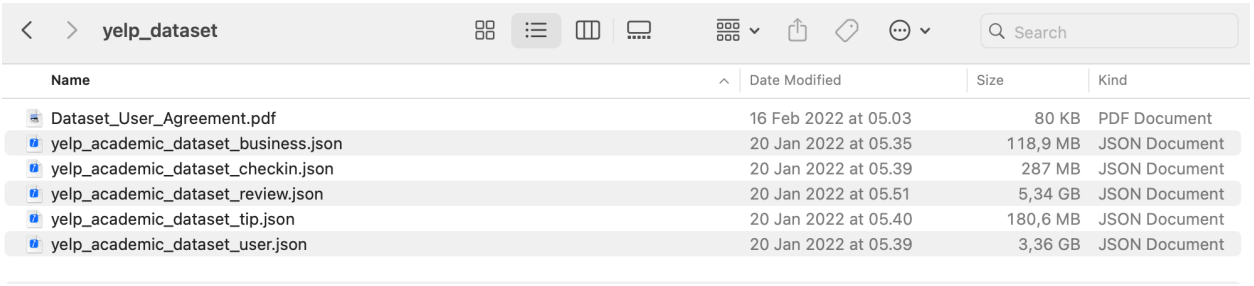
- JSON to CSV Converter: Polars
    - Polars provides memory-efficient processing of large datasets through its lazy evaluation. Given the massive size of Yelp JSON files (potentially GBs), Polars can handle the conversion without loading entire datasets into memory, preventing out-of-memory errors that could occur with pandas on large files.
- Storage: DuckDB (data/yelp_dwh.duckdb)
    - DuckDB is ideal for local analytical processing as it combines the simplicity of SQLite with the performance of columnar databases. It requires no server setup, handles large datasets efficiently, and provides excellent SQL analytics capabilities.
- Control: Makefile commands + Python CLI (main.py) + YAML config
    - This combination provides simplicity and maintainability:
        - Makefile: Offers simple, declarative pipeline orchestration (make convert → validate → load → ods → dwh)
        - Python CLI: Provides flexible scripting capabilities for complex data transformations.
        - YAML Config: Enables easy configuration management without code changes, making the pipeline adaptable to different environments and datasets.
- Transforms: Layer-specific SQL scripts (sql/ods/, sql/dwh/)
- Rollback: Drop scripts for each layer (sql/rollback/)

Star Schema (DWH Layer)

- Facts: fact_reviews (review metrics)
- Dimensions: dim_date, dim_weather, dim_business
- Analysis: Weather impact on review patterns

Extract .tar

Screen capture:



yelp_dataset

Name	Date Modified	Size	Kind
Dataset_User_Agreement.pdf	16 Feb 2022 at 05.03	80 KB	PDF Document
yelp_academic_dataset_business.json	20 Jan 2022 at 05.35	118,9 MB	JSON Document
yelp_academic_dataset_checkin.json	20 Jan 2022 at 05.39	287 MB	JSON Document
yelp_academic_dataset_review.json	20 Jan 2022 at 05.51	5,34 GB	JSON Document
yelp_academic_dataset_tip.json	20 Jan 2022 at 05.40	180,6 MB	JSON Document
yelp_academic_dataset_user.json	20 Jan 2022 at 05.39	3,36 GB	JSON Document

Convert JSON to CSV

Github commit:
https://github.com/skatesqueaker/dana_sdqe/commit/7596da4494b19b3b7cf34d19230719a820b3a898

Capture:

```
❯ make convert
python main.py convert
Converting JSON files to CSV...
2025-08-14 23:26:16,198 - INFO - Processing 5 conversions
2025-08-14 23:26:16,198 - INFO - [1/5] Processing: data/input/yelp_dataset/yelp_academic_dataset_business.json
2025-08-14 23:26:16,198 - INFO - Starting conversion: data/input/yelp_dataset/yelp_academic_dataset_business.json → data/output/yelp_business.csv
2025-08-14 23:26:18,623 - INFO - Conversion completed successfully
2025-08-14 23:26:18,628 - INFO - [2/5] Processing: data/input/yelp_dataset/yelp_academic_dataset_checkin.json
2025-08-14 23:26:18,628 - INFO - Starting conversion: data/input/yelp_dataset/yelp_academic_dataset_checkin.json → data/output/yelp_checkin.csv
2025-08-14 23:26:19,789 - INFO - Conversion completed successfully
2025-08-14 23:26:19,790 - INFO - [3/5] Processing: data/input/yelp_dataset/yelp_academic_dataset_review.json
2025-08-14 23:26:19,790 - INFO - Starting conversion: data/input/yelp_dataset/yelp_academic_dataset_review.json → data/output/yelp_review.csv
2025-08-14 23:28:18,105 - INFO - Conversion completed successfully
2025-08-14 23:28:18,354 - INFO - [4/5] Processing: data/input/yelp_dataset/yelp_academic_dataset_tip.json
2025-08-14 23:28:18,361 - INFO - Starting conversion: data/input/yelp_dataset/yelp_academic_dataset_tip.json → data/output/yelp_tip.csv
2025-08-14 23:28:21,113 - INFO - Conversion completed successfully
2025-08-14 23:28:21,115 - INFO - [5/5] Processing: data/input/yelp_dataset/yelp_academic_dataset_user.json
2025-08-14 23:28:21,116 - INFO - Starting conversion: data/input/yelp_dataset/yelp_academic_dataset_user.json → data/output/yelp_user.csv
2025-08-14 23:29:29,147 - INFO - Conversion completed successfully
2025-08-14 23:29:29,268 - INFO - All conversions completed
```

Average of around 3 minutes to convert 5 Yelp JSON dataset to CSV.

```
❯ make validate
python main.py validate
Validating conversion results...
2025-08-14 23:32:17,867 - INFO - Validating 5 conversions from config/config.yaml

2025-08-14 23:32:18,285 - INFO - [yelp_business] Records match: 150,346
2025-08-14 23:32:19,127 - INFO - [yelp_checkin] Records match: 131,930
2025-08-14 23:32:41,728 - INFO - [yelp_review] Records match: 6,990,280
2025-08-14 23:32:42,682 - INFO - [yelp_tip] Records match: 908,915
2025-08-14 23:33:02,689 - INFO - [yelp_user] Records match: 1,987,897

2025-08-14 23:33:02,762 - INFO - Validation complete: 5/5 conversions valid
2025-08-14 23:33:02,763 - INFO - Record count checks passed
```

Average of around 40 seconds to validate converted JSON with CSV source.

```
❯ make load
python main.py load
Loading CSV data into staging database...
2025-08-15 00:18:53,555 - INFO - Connected to DuckDB: data/yelp_dwh.duckdb
2025-08-15 00:18:53,561 - INFO - Created tables from sql/ddl/create_staging_tables.sql
2025-08-15 00:18:53,561 - INFO - Loading yelp_business...
2025-08-15 00:18:54,693 - INFO - Loaded 150,346 rows into staging.yelp_business
2025-08-15 00:18:54,693 - INFO - Loading yelp_checkin...
2025-08-15 00:18:55,342 - INFO - Loaded 131,930 rows into staging.yelp_checkin
2025-08-15 00:18:55,342 - INFO - Loading yelp_review...
2025-08-15 00:19:41,258 - INFO - Loaded 6,990,280 rows into staging.yelp_review
2025-08-15 00:19:41,308 - INFO - Loading yelp_tip...
2025-08-15 00:19:42,649 - INFO - Loaded 908,915 rows into staging.yelp_tip
2025-08-15 00:19:42,649 - INFO - Loading yelp_user...
2025-08-15 00:20:51,340 - INFO - Loaded 1,987,897 rows into staging.yelp_user
2025-08-15 00:20:51,366 - INFO - Loading us_weather_precipitation...
2025-08-15 00:20:51,530 - INFO - Loaded 10,137 rows into staging.us_weather_precipitation
2025-08-15 00:20:51,530 - INFO - Loading us_weather_temperature...
2025-08-15 00:20:51,556 - INFO - Loaded 10,227 rows into staging.us_weather_temperature
2025-08-15 00:20:51,556 - INFO - Loading complete: 7 tables, 10,189,732 total rows
```

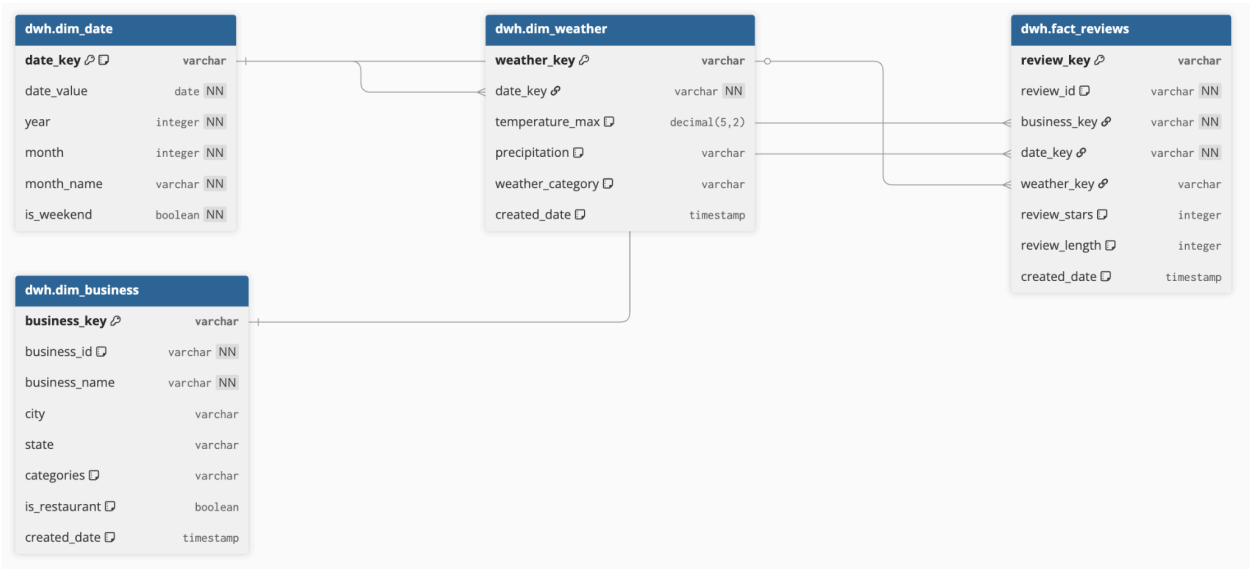
Average of 2 minutes to load CSV datasets to DuckDB database.

ODS Entity-Relationship (ER)


```
❯ make ods
python main.py ods
Transforming staging data to ODS...
2025-08-15 00:32:20,038 - INFO - Connected to DuckDB for ODS transformation: data/yelp_dwh.duckdb
2025-08-15 00:32:20,042 - INFO - Created tables from sql/ddl/create_ods_tables.sql
2025-08-15 00:32:20,042 - INFO - Running ODS transformation: sql/ods/transform_business.sql
2025-08-15 00:32:21,091 - INFO - Running ODS transformation: sql/ods/transform_user.sql
2025-08-15 00:32:54,301 - INFO - Running ODS transformation: sql/ods/transform_review.sql
2025-08-15 00:34:36,780 - INFO - Running ODS transformation: sql/ods/transform_tip.sql
2025-08-15 00:34:37,886 - INFO - Running ODS transformation: sql/ods/transform_checkin.sql
2025-08-15 00:34:41,290 - INFO - Running ODS transformation: sql/ods/transform_weather_precipitation.sql
2025-08-15 00:34:41,360 - INFO - Running ODS transformation: sql/ods/transform_weather_temperature.sql
2025-08-15 00:34:41,375 - INFO - ODS transformation complete: 7/7 transforms executed
```

Average of 2 minutes to transform data from staging to ODS.

Star Schema Diagram



Data Architecture Diagram DDL

Github commit:
https://github.com/skatesqueaker/dana_sdqe/commit/dab7a5b5e2aacf4066d96ae17cec03b19e0ae75b

Migrate ODS to DWH

Github files:
https://github.com/skatesqueaker/dana_sdqe/blob/main/yelp_dwh/sql/dwh/transform_facts.sql
https://github.com/skatesqueaker/dana_sdqe/blob/main/yelp_dwh/sql/dwh/transform_dimensions.sql
Capture:

```
❯ make dwh
python main.py dwh
Transforming ODS data to DWH...
2025-08-15 04:44:00,810 - INFO - Connected to DuckDB for DWH transformation: data/yelp_dwh.duckdb
2025-08-15 04:44:00,816 - INFO - Created tables from sql/ddl/create_dwh_tables.sql
2025-08-15 04:44:00,816 - INFO - Running DWH transformation: sql/dwh/transform_dimensions.sql
2025-08-15 04:44:02,222 - INFO - Running DWH transformation: sql/dwh/transform_facts.sql
2025-08-15 04:44:41,425 - INFO - DWH transformation complete: 2/2 transforms executed
```

Average of 40 seconds to transform data from ODS to DWH.

How Weather Affect Yelp Review

Github commit:
https://github.com/skatesqueaker/dana_sdqe/commit/dfc6e8f10a80c97aa22915b2d9a5b7bc686ee896

Capture:

```sql
-- Do people tend to review restaurants on perfect weather days?
SELECT
    CASE WHEN w.weather_category = 'Perfect' THEN 'Perfect Weather' ELSE 'Other Weather' END as weather_type,
    COUNT(r.review_key) as review_count,
    ROUND(AVG(r.review_stars), 2) as avg_rating
FROM dwh.fact_reviews r
JOIN dwh.dim_weather w ON r.weather_key = w.weather_key
JOIN dwh.dim_business b ON r.business_key = b.business_key
WHERE b.is_restaurant = true
GROUP BY weather_type;

-- How does rain affect reviews?
SELECT
    w.weather_category,
    COUNT(r.review_key) as review_count,
    ROUND(AVG(r.review_stars), 2) as avg_rating,
    ROUND(AVG(r.review_length), 0) as avg_review_length
FROM dwh.fact_reviews r
JOIN dwh.dim_weather w ON r.weather_key = w.weather_key
GROUP BY w.weather_category;
```

Results 1 | Results 2

SELECT CASE WHEN w.weather_category = 'Perfect' | Enter a SQL expression to filter results (use Ctrl+Space)

	weather_type	review_count	avg_rating
1	Other Weather	6,603,482	3.76
2	Perfect Weather	52,347	3.77

```sql
-- Do people tend to review restaurants on perfect weather days?
SELECT
    CASE WHEN w.weather_category = 'Perfect' THEN 'Perfect Weather' ELSE 'Other Weather' END as weather_type,
    COUNT(r.review_key) as review_count,
    ROUND(AVG(r.review_stars), 2) as avg_rating
FROM dwh.fact_reviews r
JOIN dwh.dim_weather w ON r.weather_key = w.weather_key
JOIN dwh.dim_business b ON r.business_key = b.business_key
WHERE b.is_restaurant = true
GROUP BY weather_type;

-- How does rain affect reviews?
SELECT
    w.weather_category,
    COUNT(r.review_key) as review_count,
    ROUND(AVG(r.review_stars), 2) as avg_rating,
    ROUND(AVG(r.review_length), 0) as avg_review_length
FROM dwh.fact_reviews r
JOIN dwh.dim_weather w ON r.weather_key = w.weather_key
GROUP BY w.weather_category;
```

Results 1 | Results 2

SELECT w.weather_category, COUNT(r.review_key) as | Enter a SQL expression to filter results (use Ctrl+Space)

	weather_category	review_count	avg_rating	avg_review_length
1	Cold	6,952	3.75	604
2	Perfect	53,871	3.76	623
3	Mild	546,104	3.75	618
4	Hot	100,284	3.71	656
5	Rainy	6,125,566	3.75	562

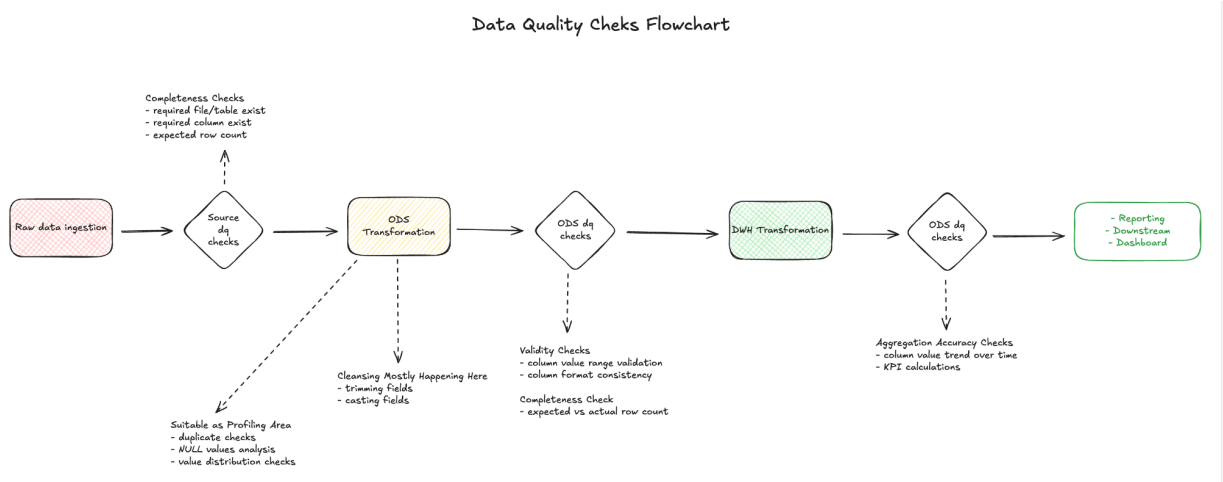
Key Insights

Consistent Behavior: People tend to rate restaurants similarly regardless of weather, indicating that food quality and service matter more than external conditions.

Hot Weather Effect: Slightly lower ratings during hot weather (3.71) might suggest customer discomfort affects satisfaction.

Business Conclusion
Weather appears to have minimal impact on Yelp restaurant ratings. This suggests that customer satisfaction is primarily driven by the restaurant experience itself rather than external weather conditions.

Data Quality Strategies



Quality Check Stages

- Source Data Checks (Entry Point)
  Completeness: Verify all required files/tables exist, columns are present, and expected row counts match.
  Purpose: Catch data delivery issues early before processing begins

- ODS Transformation Quality Gates
  Validity Checks: Ensure data values fall within expected ranges and formats are consistent.
  Consistency Checks: Verify actual vs expected row counts to detect data loss.
  Cleaning: Handle missing data, trim fields, and cast to proper data types.

- DWH Transformation Quality Gates
  Aggregation Accuracy: Validate that calculated metrics and KPIs are correct.
  Trend Analysis: Check for unusual patterns or anomalies in aggregated data over time.

Key Features

- Layered Validation: Each layer has specific quality criteria appropriate to its data maturity level (raw → cleaned → aggregated).
- Automated Pipeline: Diamond shapes represent decision points where data either passes quality checks or gets routed for cleanup/investigation.

Business Value

This strategy ensures that business users receive reliable, accurate data for decision-making while preventing poor-quality data from propagating through the system. The early detection approach saves time and prevents downstream analytical errors.

Data Quality Rule Scripts

Github commit:
https://github.com/skatesqueaker/dana_sdqe/commit/64fddd74ddbe30d5275cd0e8d588c8b9b8ee714a

DQ Checks:

Results 1

```
SELECT 'ODS' as layer, 'ods_business_null_check' as
```

Enter a SQL expression to filter results (use Ctrl+Space)

	layer	check_name	fail_count	fail_percentage	status
1	DWH	dwh_business_duplicates	0	0	PASS
2	DWH	dwh_date_consistency	0	0	PASS
3	DWH	dwh_fact_referential_integrity	0	0	PASS
4	DWH	dwh_weather_fk_consistency	0	0	PASS
5	ODS	ods_business_null_check	0	0	PASS
6	ODS	ods_checkin_key_fields	0	0	PASS
7	ODS	ods_review_validity	0	0	PASS
8	ODS	ods_tip_content_validation	0	0	PASS
9	ODS	ods_user_neg_values	0	0	PASS
10	ODS	ods_weather_precip_validation	0	0	PASS
11	ODS	ods_weather_temp_validation	0	0	PASS