

Part 3 – Database Design

Project Title: Smart Radar Traffic Monitoring System

Overview

The database design for the *Smart Radar Traffic Monitoring System* is built to efficiently handle large-scale traffic data generated by radar sensors.

It supports **real-time streaming** and **batch processing** workflows, enabling quick analysis, alerting, and visualization.

The system uses a **storage architecture**:

- 1. **Azure Data Lake** — for storing raw and processed Parquet files.
- 2. **Data Warehouse** — for structured queries and Power BI dashboards.

Database Objectives

- Store, clean, and organize incoming radar data.
- Log all types of traffic violations and real-time alerts.
- Provide fast access for analytics and dashboard queries.
- Ensure data integrity and scalability for large volumes.

Main Tables

Table Name	Description	Key Fields
vw_dim_routes	Stores static dimension data for roads, defining attributes like speed limits, center coordinates, and radar count for each route.	route_id, route_name, speed_limit, center_lat, center_lon, radar_count
vw_dim_radars	Stores static dimension data for radars, defining the location (lat/lon) and index of each radar unit on a specific route.	radar_id, route_id, radar_index, lat, lon
vw_dim_vehicles	Stores static dimension data for	plate, color, created_at

Table Name	Description	Key Fields
	vehicles, using the license plate as the primary key.	
vw_fact_journeys	Fact table storing summaries of completed trips, including total distance, total violations, and total fines incurred during the journey.	journey_id, plate, route_id, driver_profile, start_time, end_time, total_distance, total_violations, total_fines
vw_radar_logs	Stores every instance of a vehicle passing a radar. It captures speed, potential violation details (seat belt, phone usage), and location at the time of detection.	id, journey_id, plate, speed, speed_limit, color, radar_id, lat, lon, seat_belt, phone_usage, is_violation, total_fine, timestamp
vw_violations	Records detected violations separately, detailing the reason, the fine amount, and the timestamp of the infraction.	journey_id, plate, reason, fine, timestamp
vw_system_logs	Operational table used for internal system logging, capturing messages, errors, and system events.	log_level, message, details, timestamp

### Relationships (ERD Summary)

Relationship	Type	Description
One Vehicle -> Many Journeys	1:N	Each entry in vw_dim_vehicles (identified by plate) can be linked to <b>multiple</b> records in vw_fact_journeys.
One Journey -> Many Radar Logs	1:N	Each journey in vw_fact_journeys (identified by journey_id) generates <b>multiple</b> speed/detection events recorded in vw_radar_logs.
One Journey -> Many Violations	1:N	A single journey in vw_fact_journeys can result in <b>multiple</b> recorded violations in the vw_violations table.
One Route -> Many Journeys	1:N	Each route in vw_dim_routes (identified by route_id) can host <b>many</b> recorded journeys in vw_fact_journeys.
One Route -> Many Radars	1:N	Each route in vw_dim_routes (identified by route_id) has <b>multiple</b> associated radars recorded in vw_dim_radars.

Relationship	Type	Description
One Radar -> Many Radar Logs	1:N	Each physical radar in vw_dim_radars (identified by radar_id) generates <b>many</b> detection records in vw_radar_logs.

ERD Structure

