**RDBMS vs Graph — Text-Based Diagrams (Editable)**

*Generated: 2025-08-31 21:06 UTC*

**A) RDBMS schema (tables, PK/FK)**

+--------------------+ 1 \* +--------------------+  
| vehicle\_model |<-------------------------| trim |  
+--------------------+ +--------------------+  
| PK id | | PK id |  
| UQ code | | FK model\_id |  
| name | | model\_year (int) |  
+--------------------+ +--------------------+  
  
 \* \*   
+--------------------+ <--------------------> +---------------------+  
| trim | trim\_feature | feature |  
+--------------------+ ---------------------> +---------------------+  
| PK id | PK (trim\_id,feature) | PK id |  
| ... | FK trim\_id ---------->| UQ code |  
+--------------------+ FK feature\_id ------->| pref\_label |  
 +---------------------+  
  
+--------------------+ \* 1 +--------------------+  
| sale\_record |------------------------->| region |  
+--------------------+ +--------------------+  
| PK id | | PK id |  
| FK trim\_id | | UQ code |  
| FK region\_id | | name |  
| sale\_date (date) | +--------------------+  
| quantity (int) |  
+--------------------+

**B) Graph (RDF) — ontology/schema view (classes & properties)**

( VehicleModel ) <-- belongsTo -- ( Trim ) -- hasFeature --> ( Feature )  
 | |  
 | code : string | modelYear : gYear  
 v v  
 [literal] [literal]  
  
( SaleRecord ) -- forTrim --> ( Trim )  
 | \  
 | \  
 | -- soldIn --> ( Region )  
 |  
 +-- saleDate : date  
 +-- quantity : integer  
  
( Region )  
 |  
 +-- code : string

**C) Graph (RDF) — instance view (sample data)**

( ex:M1 : VehicleModel )  
 |  
 +-- code = "FALCON\_X"  
  
( ex:T101 : Trim )  
 | \  
 | \ hasFeature  
belongsTo \  
 | \--> ( ex:F\_ACC : Feature )  
 | |  
 | +-- prefLabel = "Adaptive Cruise Control"  
 |  
 +-- modelYear = 2023 (gYear)  
  
( ex:S1 : SaleRecord )  
 | \  
 | \ soldIn  
 forTrim \  
 | \--> ( ex:CA : Region )  
 |  
 +-- saleDate = 2023-03-15 (date)  
 +-- quantity = 120 (int)

**D) “Ask the same question” — join path vs graph path**

RDBMS join path (tables):  
vehicle\_model --(id)= model\_id--> trim --(id)= trim\_id--> sale\_record --(region\_id)=id--> region  
 | ^  
 | |  
 +--(id)=trim\_id--> trim\_feature --(feature\_id)=id--> feature  
Filters:  
 m.code='FALCON\_X' t.model\_year=2023 f.code='ACC' r.code='CA' s.sale\_date='2023-03-15'  
Aggregate:  
 SUM(s.quantity) GROUP BY s.trim\_id

Graph traversal path (edges):  
VehicleModel(code="FALCON\_X")  
 ^ belongsTo  
 |   
 Trim(modelYear=2023) -- hasFeature --> Feature(prefLabel/alt="ACC")  
 |  
 | forTrim  
 v  
 SaleRecord(saleDate=2023-03-15, quantity) -- soldIn --> Region(code="CA")  
Aggregate:  
 SUM(quantity) GROUP BY Trim

**E) Legend (RDBMS ↔ Graph)**

RDBMS Graph (RDF)  
--------------------------- ------------------------------  
Table Class (node type)  
Row Node (instance)  
Column Property (literal on node/edge)  
FK + JOIN Edge (predicate) + pattern match  
CHECK/NOT NULL/FK SHACL shape (constraints)  
SQL SPARQL (RDF) / Cypher (Neo4j)