Transport and Logistics Management System

Database Normalization Process

DBMS Lab Project
MIT Manipal, CCEB

Submitted by:
Anvita Warjri (230953100)
Riddhima Jain (230953108)
Adriteyo Das (230953244)

Contents

1 Functional Dependencies

Key Functional Dependencies

Primary Key, **Foreign Key**

The following functional dependencies were identified in the original schema:

- Users: user_id → username, full_name, email, phone, user_type, status, created_at, last_login, password
- Admins: admin_id → user_id, access_level
- Customers: customer_id → user_id, company_name, tax_id, credit_limit, payment_terms
- Drivers: driver_id → user_id, license_number, license_expiry, medical_check_date, training_certification, status
- Locations: location_id → address, city, state, country, postal_code, latitude, longitude, location_type
- Warehouses: warehouse_id → location_id, warehouse_name, capacity, current_occupancy, manager_id, operating_hours
- Vehicles: vehicle_id → license_plate, make, model, year, capacity_kg, vehicle_type, status, current_location_id, last_inspection_date
- Routes: route_id → route_name, origin_id, destination_id, distance_km, estimated_duration_min, status, hazard_level
- Waypoints: waypoint_id → route_id, location_id, sequence_number, estimated_arrival, estimated_departure
- Shipments: shipment_id → tracking_number, customer_id, origin_id, destination_id, route_id, vehicle_id, driver_id, status, total_weight, total_volume, shipment_value, insurance_required, special_instructions, created_at, pickup_date, estimated_delivery, actual_delivery
- Shipment_Items: item_id → shipment_id, description, quantity, weight, volume, item_value, is_hazardous, is_fragile
- Tracking_Events: event_id → shipment_id, event_type, location_id, event_timestamp, recorded_by, notes

2 Original Database Schema

The initial database consists of 12 tables with various functional dependencies. Each table will be analyzed and normalized through the four normal forms.

2.1 Original Tables

Users (Original)

Field	**Type**
user_id	int
username	varchar(50)
full_name	varchar(100)
email	varchar(100)
phone	varchar(15)
user_type	enum('admin','driver','customer')
status	enum('active','inactive')
created_at	timestamp
last_login	timestamp
password	text

Admins (Original)

Field	**Type**
admin_id	int
user_id	int
access_level	enum('super','regular')

Customers (Original)

Field	**Type**
customer_id	int
user_id	int
company_name	varchar(100)
tax_id	varchar(20)
credit_limit	decimal(10,2)
payment_terms	varchar(50)

Drivers (Original)

Field	**Type**
driver_id	int
user_id	int
license_number	varchar(20)
license_expiry	date
medical_check_date	date
training_certification	varchar(100)
status	enum(’available’,’on_leave’,’assigned’)

Locations (Original)

Field	**Type**
location_id	int
address	varchar(255)
city	varchar(50)
state	varchar(50)
country	varchar(50)
postal_code	varchar(10)
latitude	decimal(10,8)
longitude	decimal(11,8)
location_type	enum(’warehouse’,’customer’,’drop_point’)

Warehouses (Original)

Field	**Type**
warehouse_id	int
location_id	int
warehouse_name	varchar(100)
capacity	decimal(10,2)
current_occupancy	decimal(10,2)
manager_id	int
operating_hours	varchar(100)

Vehicles (Original)

Field	Type
vehicle_id	int
license_plate	varchar(15)
make	varchar(50)
model	varchar(50)
year	int
capacity_kg	decimal(10,2)
vehicle_type	enum('truck','van','trailer','pickup')
status	enum('available','in_maintenance','in_use')
current_location_id	int
last_inspection_date	date

Routes (Original)

Field	Type
route_id	int
route_name	varchar(100)
origin_id	int
destination_id	int
distance_km	decimal(6,2)
estimated_duration_min	int
status	enum('active','inactive')
hazard_level	enum('low','medium','high')

Waypoints (Original)

Field	Type
waypoint_id	int
route_id	int
location_id	int
sequence_number	int
estimated_arrival	time
estimated_departure	time

Shipments (Original)

Field	**Type**
shipment_id	int
tracking_number	varchar(20)
customer_id	int
origin_id	int
destination_id	int
route_id	int
vehicle_id	int
driver_id	int
status	enum('pending','picked_up','in_transit','delivered','returned')
total_weight	decimal(10,2)
total_volume	decimal(10,2)
shipment_value	decimal(12,2)
insurance_required	tinyint(1)
special_instructions	text
created_at	timestamp
pickup_date	datetime
estimated_delivery	datetime
actual_delivery	datetime

Shipment_Items (Original)

Field	**Type**
item_id	int
shipment_id	int
description	varchar(255)
quantity	int
weight	decimal(10,2)
volume	decimal(10,2)
item_value	decimal(10,2)
is_hazardous	tinyint(1)
is_fragile	tinyint(1)

Tracking_Events (Original)

Field	**Type**
event_id	int
shipment_id	int
event_type	enum('pickup','departure','arrival','delivery','delay','issue')
location_id	int
event_timestamp	timestamp
recorded_by	int
notes	text

3 Normalization Process

3.1 First Normal Form (1NF)

1NF Requirements

- Each table cell must contain atomic values
- Each record must be unique
- No repeating groups

3.1.1 1NF Violations Found

- `Warehouses.operating_hours` - Could contain multiple values (e.g., "Mon-Fri 9-5, Sat 10-2")
- `Drivers.training_certification` - Could contain multiple certifications
- `Shipments.special_instructions` - Could contain multiple instructions

3.1.2 1NF Transformations

1. Created `Warehouse_Operating_Hours` table to store individual operating hours
2. Created `Driver_Certifications` table for multiple certifications
3. Created `Shipment_Instructions` table for multiple instructions

Warehouse_Operating_Hours (New)

Field	Type
operating_hours_id	int
warehouse_id	int
day_of_week	enum('mon','tue','wed','thu','fri','sat','sun')
opening_time	time
closing_time	time

Driver_Certifications (New)

Field	Type
certification_id	int
driver_id	int
certification_name	varchar(100)
issuing_authority	varchar(100)
issue_date	date
expiry_date	date

Shipment_Instructions (New)

Field	Type
instruction_id	int
shipment_id	int
instruction_type	enum('handling','delivery','special')
details	text
priority	enum('low','medium','high')

3.1.3 Modified Tables After 1NF

Warehouses (Modified)

Field	Type
warehouse_id	int
location_id	int
warehouse_name	varchar(100)
capacity	decimal(10,2)
current_occupancy	decimal(10,2)
manager_id	int

Drivers (Modified)

Field	Type
driver_id	int
user_id	int
license_number	varchar(20)
license_expiry	date
medical_check_date	date
status	enum('available','on_leave','assigned')

Shipments (Modified)

Field	Type
shipment_id	int
tracking_number	varchar(20)
customer_id	int
origin_id	int
destination_id	int
route_id	int
vehicle_id	int
driver_id	int
status	enum('pending','picked_up','in_transit','delivered','returned')
total_weight	decimal(10,2)
total_volume	decimal(10,2)
shipment_value	decimal(12,2)
insurance_required	tinyint(1)
created_at	timestamp
pickup_date	datetime
estimated_delivery	datetime
actual_delivery	datetime

3.2 Second Normal Form (2NF)

2NF Requirements

- Must be in 1NF
- No partial dependencies (all non-key attributes depend on entire primary key)

3.2.1 2NF Violations Found

- In Waypoints, sequence_number depends only on route_id (partial dependency)
- In Tracking_Events, location_id depends on both shipment_id and event_type

3.2.2 2NF Transformations

1. Created Route_Sequence table to handle sequence numbers
2. Modified Tracking_Events to remove partial dependencies

Route_Sequence (New)

Field	Type
route_id	int
sequence_count	int
is_circular	tinyint(1)

Waypoints (Modified)

Field	Type
waypoint_id	int
route_id	int
location_id	int
estimated_arrival	time
estimated_departure	time

Tracking_Events (Modified)

Field	Type
event_id	int
shipment_id	int
event_type	enum('pickup','departure','arrival','delivery','delay','issue')
event_timestamp	timestamp
recorded_by	int
notes	text

Event_Locations (New)

Field	Type
event_id	int
location_id	int

3.3 Third Normal Form (3NF)

3NF Requirements

- Must be in 2NF
- No transitive dependencies (non-key attributes must depend only on the primary key)

3.3.1 3NF Violations Found

- In `Locations`, `city`, `state`, `country` depend on `postal_code`
- In `Vehicles`, `make`, `model`, `year` have dependencies
- In `Users`, `email` and `username` can determine other attributes

3.3.2 3NF Transformations

1. Created `Postal_Regions` table
2. Created `Vehicle_Models` table
3. Restructured `Users` table

Postal_Regions (New)

Field	Type
postal_code	varchar(10)
city	varchar(50)
state	varchar(50)
country	varchar(50)

Locations (Modified)

Field	**Type**
location_id	int
address	varchar(255)
postal_code	varchar(10)
latitude	decimal(10,8)
longitude	decimal(11,8)
location_type	enum('warehouse','customer','drop_point')

Vehicle_Models (New)

Field	**Type**
model_id	int
make	varchar(50)
model	varchar(50)
year	int
capacity_kg	decimal(10,2)
vehicle_type	enum('truck','van','trailer','pickup')

Vehicles (Modified)

Field	**Type**
vehicle_id	int
license_plate	varchar(15)
model_id	int
status	enum('available','in_maintenance','in_use')
current_location_id	int
last_inspection_date	date

User_Credentials (New)

Field	**Type**
user_id	int
username	varchar(50)
password	text
last_login	timestamp

User_Contact (New)

Field	**Type**
user_id	int
email	varchar(100)
phone	varchar(15)

Users (Modified)

Field	**Type**
user_id	int
full_name	varchar(100)
user_type	enum('admin','driver','customer')
status	enum('active','inactive')
created_at	timestamp

3.4 Boyce-Codd Normal Form (BCNF)

BCNF Requirements

- Must be in 3NF
- For every functional dependency X → Y, X must be a superkey

3.4.1 BCNF Violations Found

- In `Drivers`, `license_number` is unique and can determine `driver_id`
- In `Shipments`, `tracking_number` is unique and can determine `shipment_id`

3.4.2 BCNF Transformations

1. Created `Driver_Licenses` table
2. Created `Shipment_Tracking` table

Driver_Licenses (New)

Field	**Type**
license_number	varchar(20)
driver_id	int
license_expiry	date
issuing_authority	varchar(100)

Drivers (Modified)

Field	Type
driver_id	int
user_id	int
medical_check_date	date
status	enum(’available’,’on_leave’,’assigned’)

Shipment_Tracking (New)

Field	Type
tracking_number	varchar(20)
shipment_id	int
issued_date	timestamp

Shipments (Modified)

Field	Type
shipment_id	int
customer_id	int
origin_id	int
destination_id	int
route_id	int
vehicle_id	int
driver_id	int
status	enum(’pending’,’picked_up’,’in_transit’,’delivered’,’returned’)
total_weight	decimal(10,2)
total_volume	decimal(10,2)
shipment_value	decimal(12,2)
insurance_required	tinyint(1)
created_at	timestamp
pickup_date	datetime
estimated_delivery	datetime
actual_delivery	datetime

4 Final Normalized Schema

After complete normalization through BCNF, the database now consists of 22 tables with proper relationships and minimal redundancy.

4.1 Complete List of Normalized Tables

1. Users
2. User_Credentials
3. User_Contact
4. Admins
5. Customers
6. Drivers
7. Driver_Licenses
8. Driver_Certifications
9. Postal_Regions
10. Locations
11. Warehouses
12. Warehouse_Operating_Hours
13. Vehicle_Models
14. Vehicles
15. Routes
16. Route_Sequence
17. Waypoints
18. Shipments
19. Shipment_Tracking
20. Shipment_Items
21. Shipment_Instructions
22. Tracking_Events
23. Event_Locations

5 Conclusion

The normalization process successfully transformed the original database schema into a well-structured design that:

- Eliminates data redundancy
- Ensures data integrity
- Follows relational database best practices
- Provides flexibility for future modifications