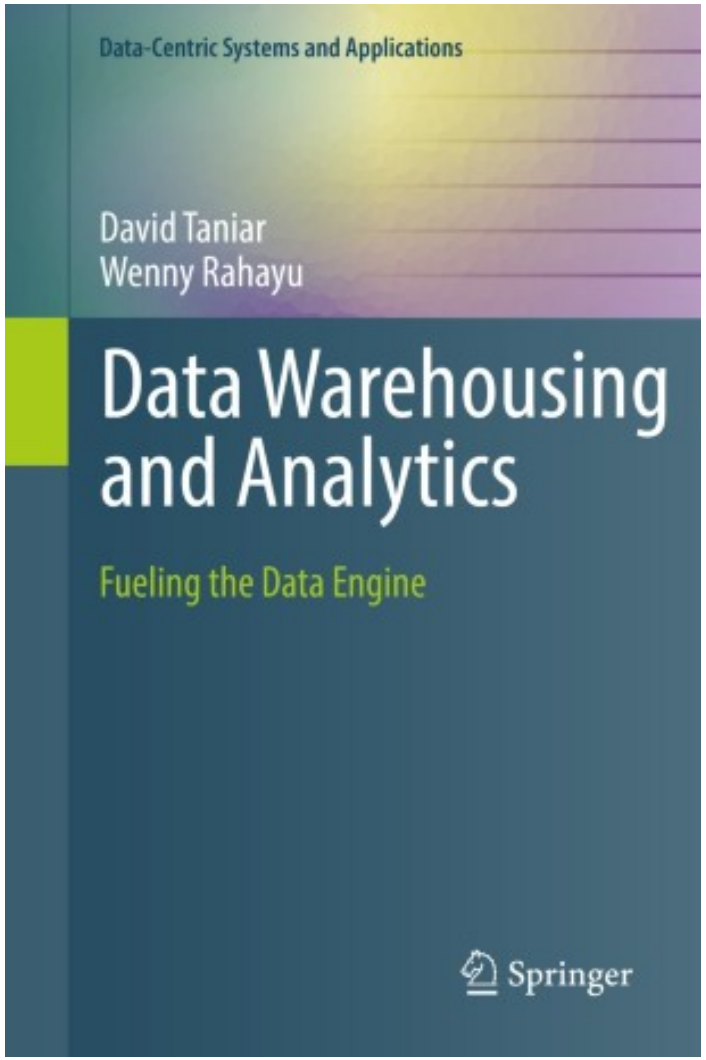


Chapter 5
Bridge Tables

Overview

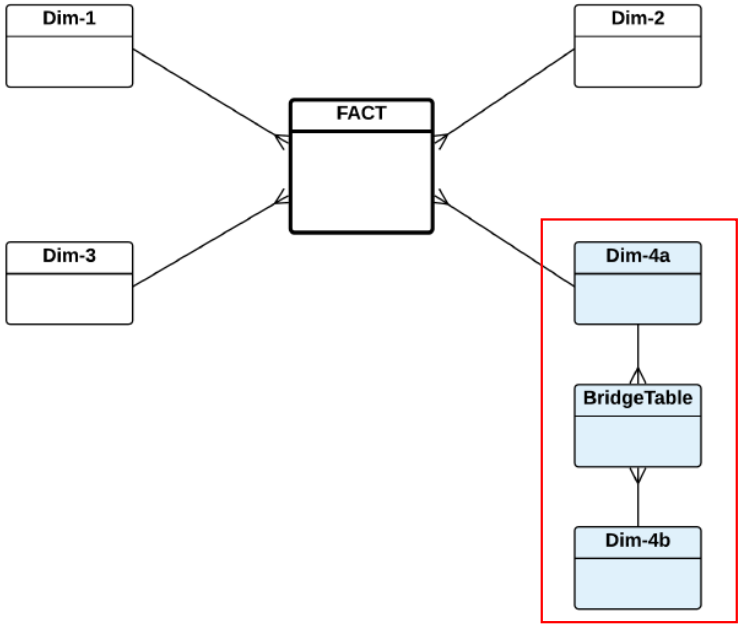
- Definition of bridge tables
- How and when to use bridge table
- Snowflakes schema in Data Warehousing

Bridge Tables

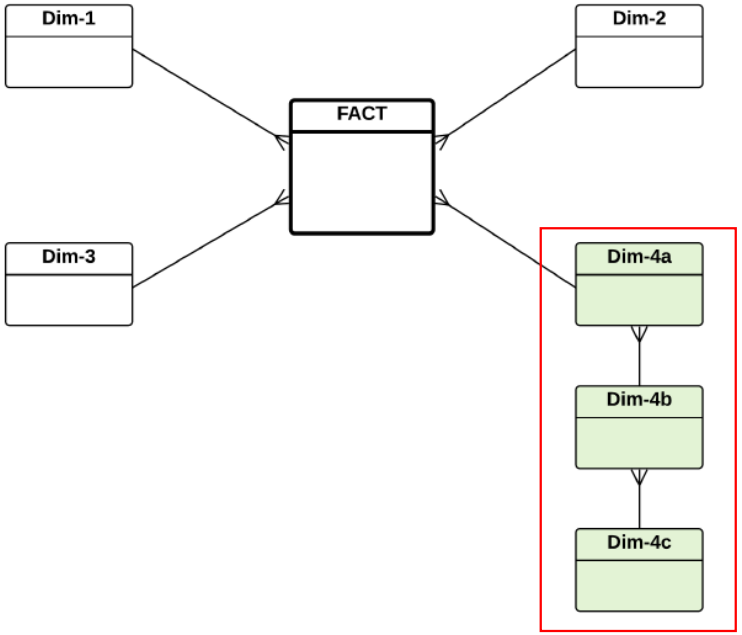
- A bridge table is a table that links between two dimensions; and only one of these two dimensions are linked to the fact.
- The star schema becomes a snowflake schema
- Bridge table is also a snowflake
- The relationship between the two dimensions that are linked through a bridge table has a cardinality of 1-many and many-1

Bridge Tables

Snowflake with a Bridge Table



Snowflake with a Hierarchy

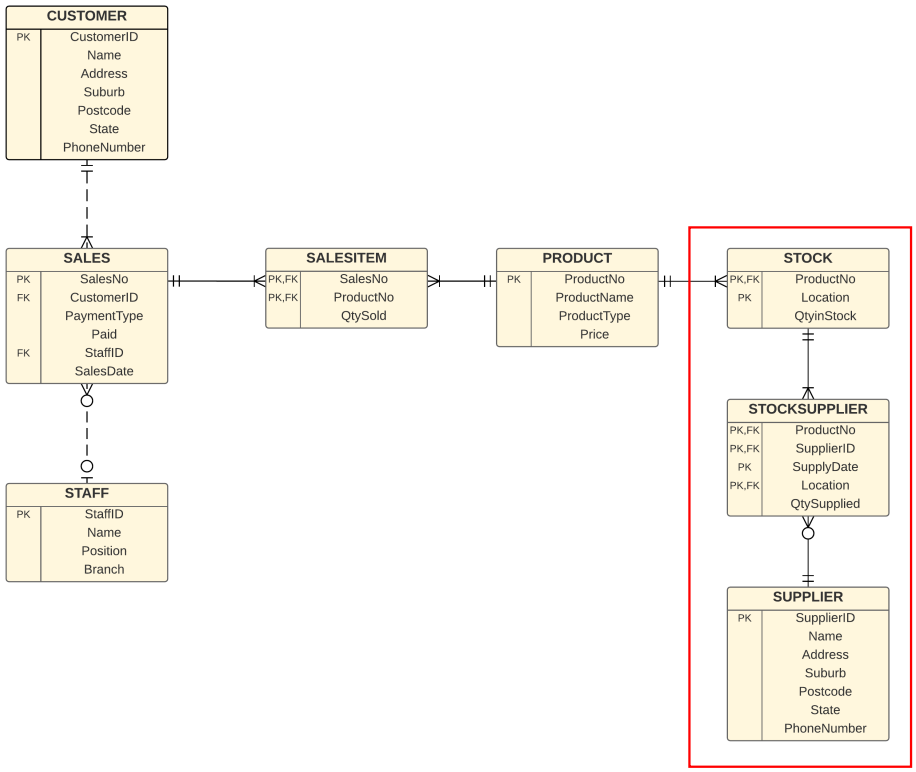


Bridge Tables

- There are at least two reasons, why a dimension cannot be connected directly to the Fact:
  - The Fact table has a fact measure, and the dimension has a key identity
  - The operational database does not have this data if the relationship between two entities in the operational database that hold the information about dimension's key identity and the intended fact measure is a *many-many* relationship.

1. A Product Sales Case Study

- Analyse the statistics of its product sales history
- Analysing the *total sales* (*quantity * price*) by
  1. *product*
  2. *customer suburbs*
  3. *sales time periods* (month and year)
  4. *supplier*



1. A Product Sales Case Study

- Imaginary 2-column tables

Table 5.1 Product point of view

ProductNo	TotalSales
A1	$130,000
B2	$15,900
C3	$2,500,000
...	...

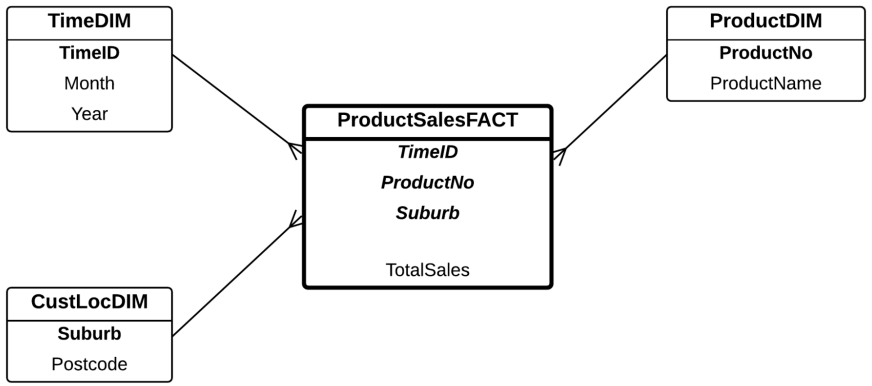
Table 5.2 Time point of view

TimeID	TotalSales
201801	$25,000
201802	$4,700
201803	$3,500
...	...

Table 5.3 Suburb point of view

Suburb	TotalSales
Caulfield	$6,500
Chadstone	$12,000
Clayton	$1,800
...	...

Initial Star Schema



- No Supplier yet

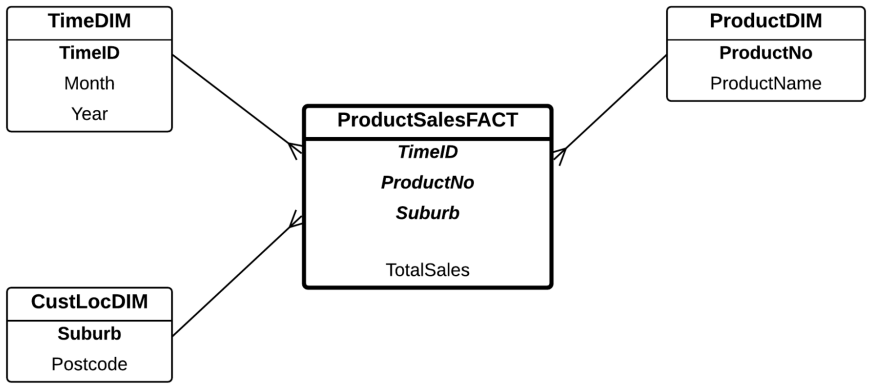
1. A Product Sales Case Study

- And the Fact table

Table 5.4 ProductSalesFact table

TimeID	Suburb	ProductNo	TotalSales
201801	Caulfield	A1	$450
201801	Caulfield	B2	$100
201801	Caulfield	C3	$320
201801	Caulfield	...	...
201801	...	...	...
201801	Chadstone	A1	$75
201801	Chadstone	B2	$600
201801	Chadstone	C3	$55
201801	Chadstone	...	...
201801	...	...	...
201801	Clayton	A1	$130
201801	...	...	...
201802	Caulfield	A1	$500
201802	Caulfield	B2	$430
201802	Caulfield	C3	$120
...	...	...	...

Initial Star Schema



- No Supplier yet

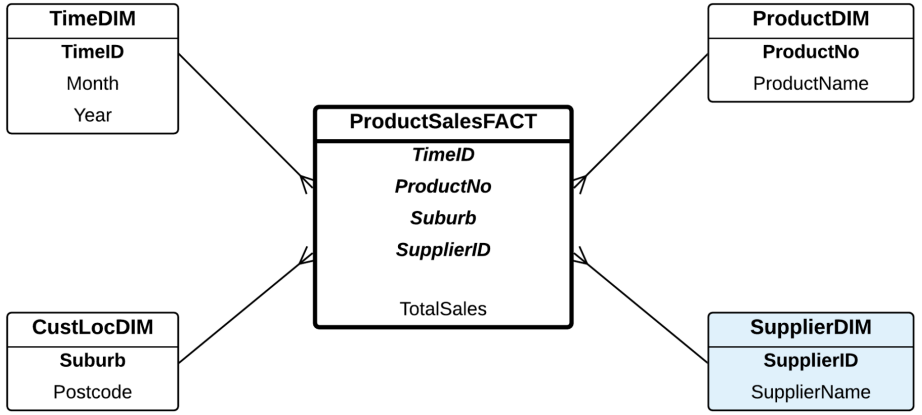
1. A Product Sales Case Study

- Now the Supplier

Table 5.5 Supplier point of view

SupplierID	TotalSales
S1	$77,000
S2	$5,700
S3	$12,500
...	...

- The new star schema with supplier
- Is this correct? How does the fact table look like?



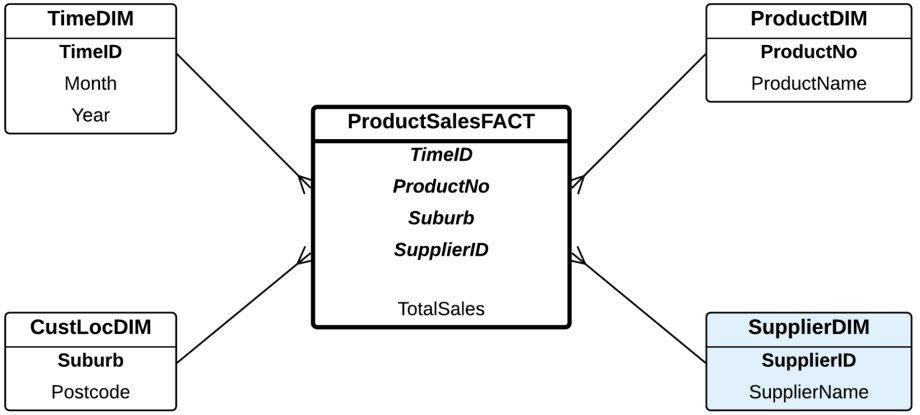
1. A Product Sales Case Study

- The new Fact table

Table 5.6 ProductSalesFact table with Supplier

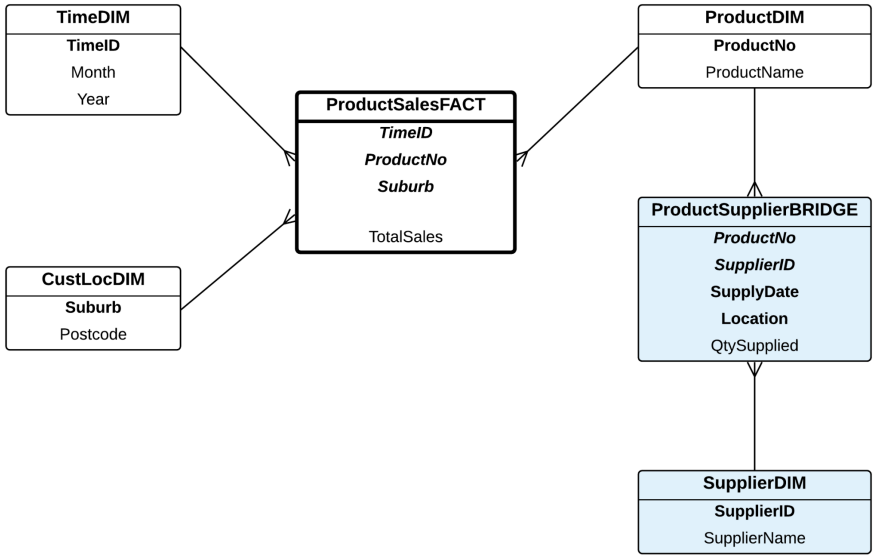
TimeID	Suburb	ProductNo	SupplierID	TotalSales
201801	Caulfield	A1	S1	...
201801	Caulfield	A1	S2	...
201801	Caulfield	A1	S3	...
201801	Caulfield	A1	...	...
201801	Caulfield	B2	S1	...
201801	Caulfield	B2	S2	...
201801	Caulfield	B2	S3	...
201801	Caulfield	B2	...	...
201801	Caulfield	C3	S1	...
201801	Caulfield	C3	S2	...
201801	Caulfield	C3	S3	...
201801	Caulfield	C3	...	...
201801	...	...	...	...
201801	Chadstone	A1	S1	...
201801	Chadstone	A1	S2	...
201801	Chadstone	A1	S3	...
201801	Chadstone	A1	...	...
201801	...	...	...	...
201802	Caulfield	A1	S1	...
201802	Caulfield	A1	S2	...
201802	Caulfield	A1	S3	...
201802	Caulfield	A1	...	...
...	...	...	...	

- The problem: there is no direct relationship between supplier and product sales
- The supplier information is not available in the sales of a particular product.
- Wrong Fact table, due to Supplier – Product being *many*-to-*many*



1. A Product Sales Case Study

Snowflake Schema with a Bridge Table



- **Bridge Table** between Product Dimension and Supplier Dimension

The SQL

```
-- Time Dimension
create table TimeDim as
select
   distinct to_char(SalesDate, 'YYYYMM') as TimeID,
   to_char(SalesDate, 'YYYY') as Year,
   to_char(SalesDate, 'MM') as Month
from Sales;

-- Customer Location Dimension
create table CustLocDim as
select distinct Suburb, Postcode
from Customer;

-- Product Dimension
create table ProductDim as
select distinct ProductNo, ProductName
from Product;
```

```
-- Bridge Table
create table ProductSupplierBridge as
select *
from StockSupplier;

-- Supplier Dimension
create table SupplierDim as
Select SupplierID, Name as SupplierName
from Supplier;

-- Fact Table
create table ProductSalesFact as
Select
   to_char(S.SalesDate, 'YYYYMM') as TimeID,
   P.ProductNo,
   C.Suburb,
   sum(SI.QtySold*P.Price) as TotalSales
from Sales S, Product P, Customer C, SalesItem SI
where S.SalesNo = SI.SalesNo
and SI.ProductNo= P.ProductNo
and C.CustomerID = S.CustomerID
group by
   to_char(S.SalesDate, 'YYYYMM'),
   P.ProductNo,
   C.Suburb;
```

1. A Product Sales Case Study

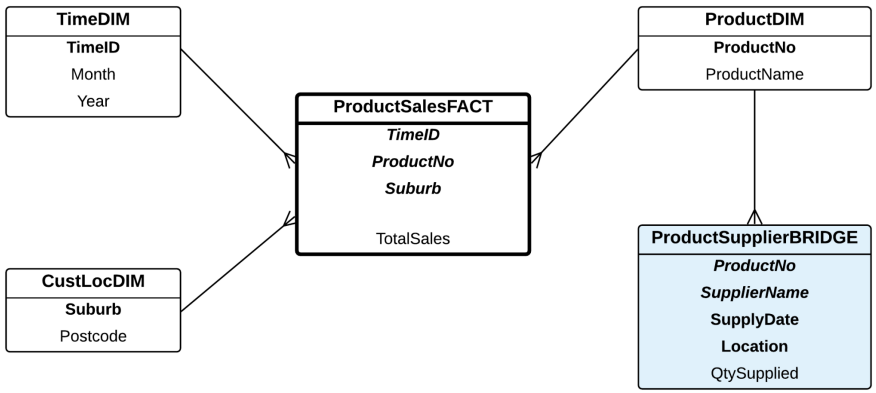
Table 5.7 ProductDim table

ProductNo	ProductName
A1	Fandy Handbag
B2	Mercer Women Shoes
C3	Plain T-Shirt
...	...

Snowflake Schema with a Bridge Table, but without a Supplier Dimension

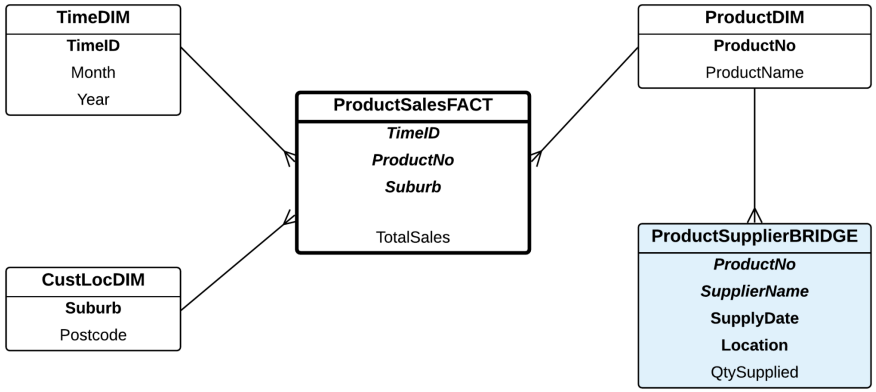
Table 5.8 ProductSupplierBridge table

ProductNo	SupplierName	DateSupplied	Location	QtySupplied
A1	Cheap Goods Pty	21-May-2018	Caulfield	100
A1	Cheap Goods Pty	15-Aug-2018	Caulfield	150
A1	Cheap Goods Pty	19-Dec-2018	Clayton	50
A1	Cheap Goods Pty	...	...	...
A1	Just Bags	21-May-2018	Chadstone	200
A1	Just Bags	30-Jun-2018	Clayton	80
A1	JustBags	...	...	...
A1	Baggy	...	...	...
A1	...	...	...	...
B2	Cheap Goods Pty	...	...	...
...	...	...	...	...

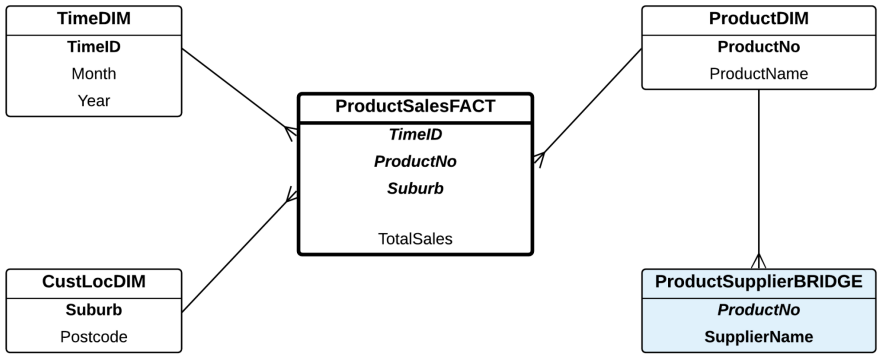


1. A Product Sales Case Study

Snowflake Schema with a Bridge Table, but <u>without a Supplier Dimension</u>



Snowflake Schema with a Bridge Table, but <u>without maintaining the history of supplies</u>



2. A Truck Delivery Case Study

- A trucking company is responsible for picking up goods from warehouses of a retail chain company, and delivering the goods to individual retail stores
  - A trip may pick up goods from many warehouses
  - A trip uses one truck only
  - A truck may have many trips
  - A trip delivers goods to several destinations

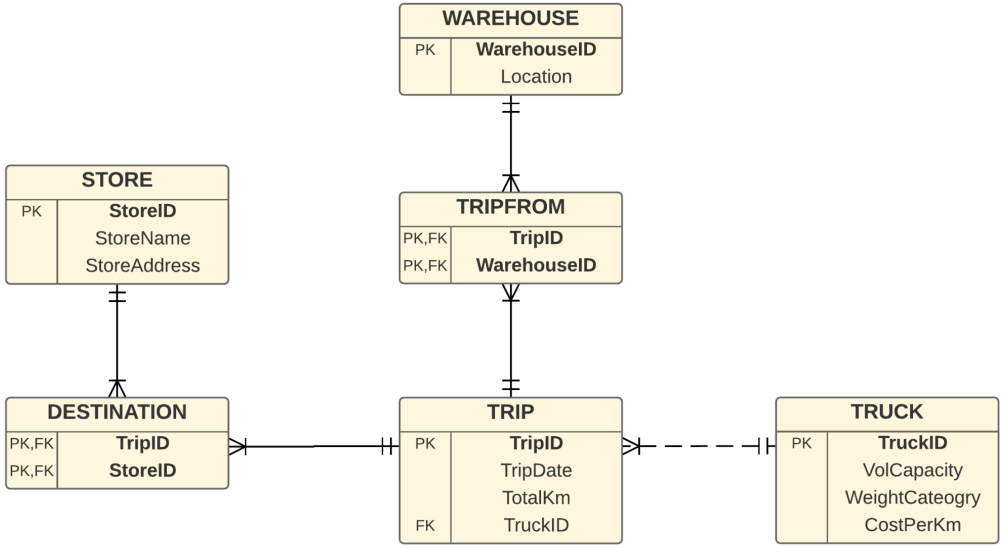


Table 5.10 Trip table

TripID	Date	TotalKm	TruckID
Trip1	14-Apr-2018	370	Truck1
Trip2	14-Apr-2018	570	Truck2
Trip3	14-Apr-2018	250	Truck3
Trip4	15-Apr-2018	450	Truck1
...	...	...	...

Table 5.11 TripFrom table

TripID	WarehouseID
Trip1	W1
Trip1	W2
Trip1	W3
Trip2	W1
Trip2	W2
...	...

Table 5.12 Truck table

TruckID	VolCapacity	WeightCategory	CostPerKm
Truck1	250	Medium	$1.20
Truck2	300	Medium	$1.50
Truck3	100	Small	$0.80
Truck4	550	Large	$2.30
Truck5	650	Large	$2.50
...	...	...	...

Table 5.13 Store table

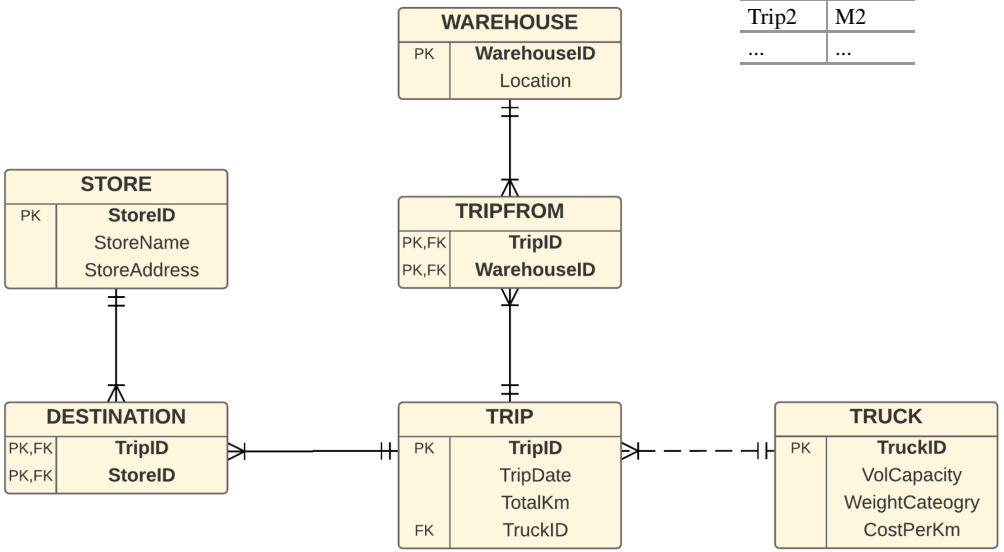
StoreID	StoreName	Address
M1	MyStore City	Melbourne
M2	MyStore Chaddy	Chadstone
M3	MyStore HiPoint	High Point
M4	MyStore Westfield	Doncaster
M5	MyStore North	Northland
M6	MyStore South	Southland
M7	MyStore East	Eastland
M8	MyStore Knox	Knox City
...	...	...

Table 5.9 Warehouse table

WarehouseID	Location
W1	Warehouse1
W2	Warehouse1
W3	Warehouse1
...	...

Table 5.14 Destination table

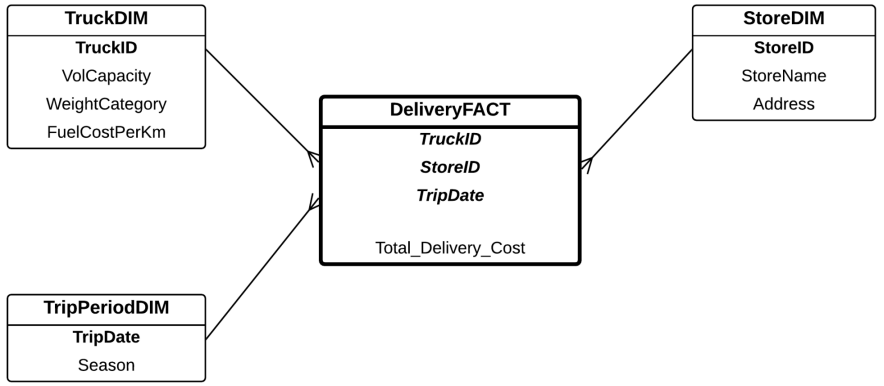
TripID	StoreID
Trip1	M1
Trip1	M2
Trip1	M4
Trip1	M3
Trip1	M8
Trip2	M4
Trip2	M1
Trip2	M2
...	...



2.1. Solution Model 1 - Using a Bridge Table

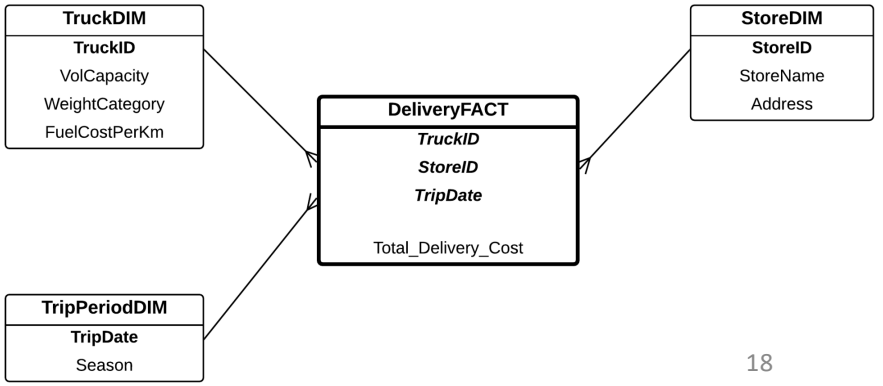
- Fact table is "Total Delivery Cost", which is calculated by *distance* and *cost per kilometre*
- The dimensions are Truck, Trip Period and Store

First attempt - Truck Delivery Star Schema



2.1. Solution Model 1 - Using a Bridge Table

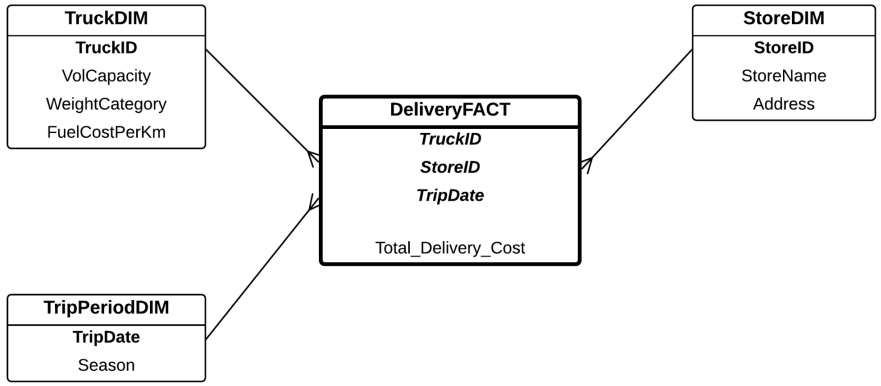
- From the Truck point of view, Truck1 makes two trips (e.g. Trip1 and Trip4), travelling a total of 820 km (370 km + 450 km). The cost for Truck1 is $1.20/km. Hence, calculating the cost for Truck1 is straightforward. The cost of the other trucks can also be calculated this way.
- From the Period point of view (say from a date point of view), three trips were made on 14 April 2018 (e.g. Trip1, Trip2 and Trip3). Trip1 (370 km) is delivered by Truck1 which costs $1.20/km. Trip2 and Trip 3 on the same day can be calculated in the same way. Hence, on 14 April 2018, the total cost can be calculated.
- From the Store point of view, the cost is calculated based on Trip, but a trip delivers goods to many stores. Therefore, the delivery cost for each store cannot be calculated. The delivery cost is for the trip, not for the store.



2.1. Solution Model 1 - Using a Bridge Table

- **Incorrect** schema, because there is **no** direct relationship between Store and total delivery cost in the Fact table

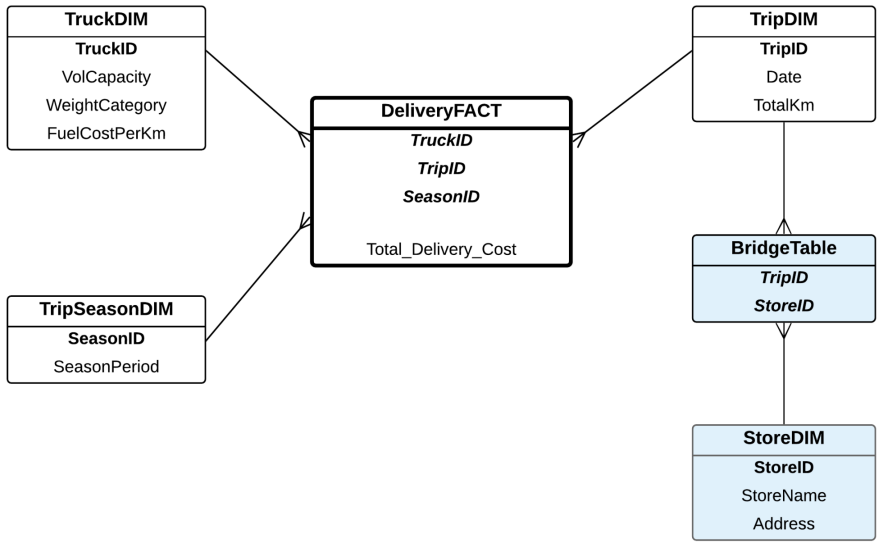
Incorrect Truck Delivery Star Schema



2.1. Solution Model 1 - Using a Bridge Table

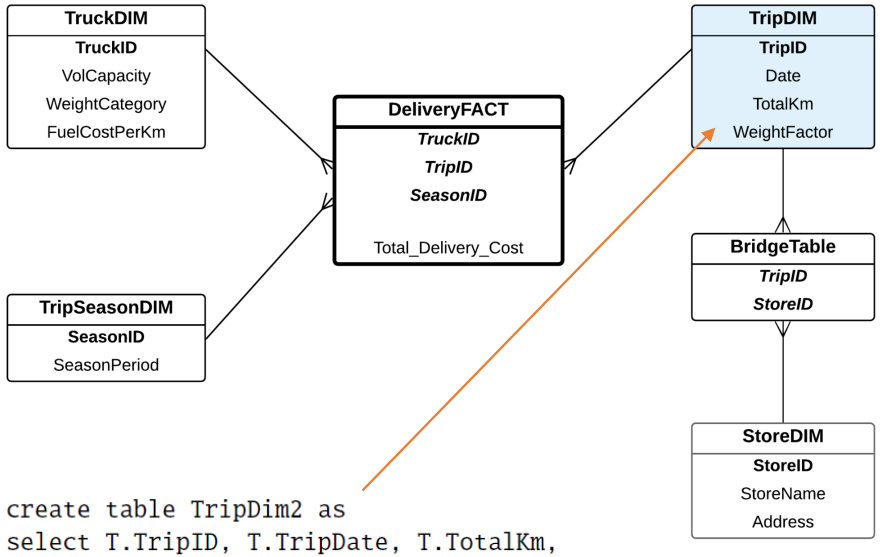
- Solve *many-many* relationship between Trip entity and Store entity
- A bridge table is used

Correct Truck Delivery Snowflake Schema with a Bridge Table



2.2. Solution Model 2 – add a Weight Factor attribute

- A weight factor is a proportion of the trip that goes to each store for that particular trip
- Estimate the contribution of each destination to the total delivery cost
- Example: 1 trip to 5 destination, each destination contributes 20% of cost




```
create table TripDim2 as
select T.TripID, T.TripDate, T.TotalKm,
   1.0/count(*) as WeightFactor
from Trip T, Destination D
where T.TripID = D.TripID
group by T.tripid, T.tripdate, T.totalkm;
```

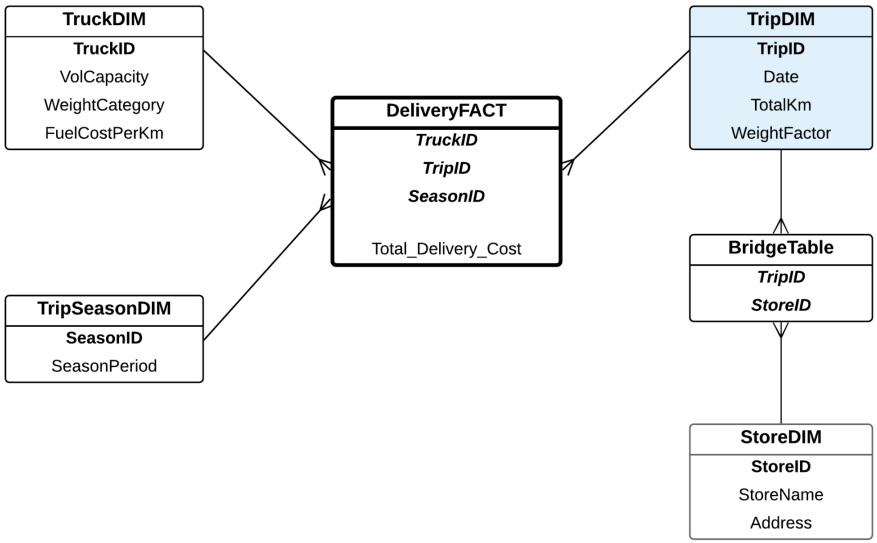
2.2. Solution Model 2 – add a Weight Factor attribute

Table 5.15 Trip Dimension table

TripID	Date	TotalKm	WeightFactor
Trip1	14-Apr-2018	370	0.20
Trip2	14-Apr-2018	570	0.33
...	...	...	...

Table 5.16 Bridge Table

TripID	StoreID
Trip1	M1
Trip1	M2
Trip1	M4
Trip1	M3
Trip1	M8
Trip2	M4
Trip2	M1
Trip2	M2
...	...



2.2. Solution Model 2 – add a Weight Factor attribute

Why store the Weight Factor attribute in the Trip Dimension?
Why not in the Bridge Dimension?

Table 5.17 Store table

StoreID	StoreName	Address
M1	MyStore City	Melbourne
M2	MyStore Chaddy	Chadstone
M3	MyStore HiPoint	High Point
M4	MyStore Westfield	Doncaster
M5	MyStore North	Northland
M6	MyStore South	Southland
M7	MyStore East	Eastland
M8	MyStore Knox	Knox City
...	...	...

Table 5.18 Trip Dimension table

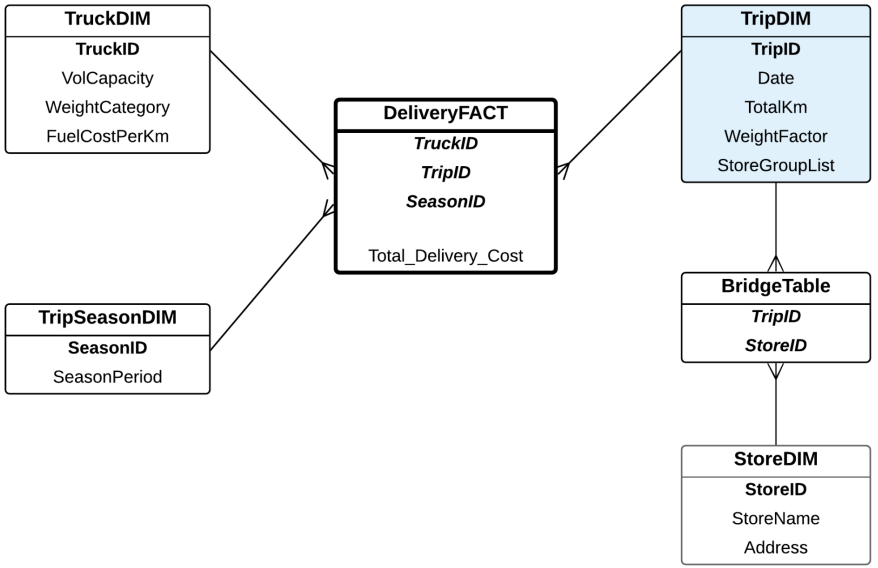
TripID	Date	TotalKm
Trip1	14-Apr-2018	370
Trip2	14-Apr-2018	570
...	...	...

Table 5.19 Bridge Table

TripID	StoreID	WeightFactor
Trip1	M1	0.20
Trip1	M2	0.20
Trip1	M4	0.20
Trip1	M3	0.20
Trip1	M8	0.20
Trip2	M4	0.33
Trip2	M1	0.33
Trip2	M2	0.33
...	...	...

2.3. Solution Model 3 - a List Aggregate version

- Adds one additional attribute in the parent dimension
- Keeps the information of the group of each record in the parent dimension table
- E.g:
  “StoreGroupList”: all stores for each trip are concatenate to become a store group list



2.3. Solution Model 3 - a List Aggregate version

- A group list is physically listed in the parent dimension
- Visually help the decision makers to understand the completeness of the group list
- Creating a List Aggregate can be very complex

```
create table TripDim3 as
select T.TripID, T.TripDate, T.TotalKm,
   1.0/count(D.StoreID) as WeightFactor,
   listagg (D.StoreID, '_') within group
   (order by D.StoreID) as StoreGroupList
from Trip T, Destination D
where T.TripID = D.TripID
group by T.TripID, T.TripDate, T.TotalKm;
```

Table 5.20 Trip Dimension table

TripID	Date	TotalKm	WeightFactor	StoreGroupList
Trip1	14-Apr-2018	370	0.20	M1_M2_M3_M4_M8
Trip2	14-Apr-2018	570	0.33	M1_M2_M4
...	...	...	...	...

Summary

- Bridge Table is used:
  - There is no relationship between this dimension and the Fact table
  - When an entity (which will become a dimension) has a *many-many* relationship with another entity (dimension) in the E/R schema
  - When temporality aspect (data history) is maintained in the operational database and the bridge table can be used to accommodate the dimension that has temporal attributes
- When a Bridge Table is used in the schema, there are two additional options:
  - A Weight Factor is used to estimate the contribution of a dimension in the calculation of the fact measure.
  - Every snowflake schema (whether it has Weight Factor or not) can be implemented in two ways: a List Aggregate version, and a non-List Aggregate version.