Data Cubes for Foodmart Analysis

Query to create Data cube “country\_wise\_sales”:

CREATE OR REPLACE VIEW country\_wise\_sales AS

select p.product\_id, p.product\_name, store.store\_country, store.store\_state, product\_family, product\_category, sf.store\_sales, sf.unit\_sales, t.time\_id, t.the\_date, t.the\_month, t.the\_year, c.customer\_id, c.gender, c.marital\_status, store.store\_type, store.store\_id,

   (case when sf.promotion\_id > 0 then 'YES' else 'NO' end) sales\_with\_promotion,

   (case when pr.promotion\_id > 0 then 'YES' else 'NO' end) product\_with\_promotion

from (select \* from sales\_fact\_1998 union select \* from sales\_fact\_dec\_1998) as sf

JOIN store on store.store\_id = sf.store\_id

join product p on p.product\_id = sf.product\_id

join product\_class pc on pc.product\_class\_id = p.product\_class\_id

join time\_by\_day t on t.time\_id = sf.time\_id

join customer c on c.customer\_id = sf.customer\_id

join promotion pr on pr.promotion\_id = sf.promotion\_id;

Creating Data cube\_”sale\_report\_complete\_summary”

CREATE OR REPLACE VIEW foodmart\_sale\_report\_complete\_summary AS

SELECT sf.store\_sales, sf.unit\_sales, (CASE WHEN sf.promotion\_id > 0 THEN 'YES' ELSE 'NO' END) AS promo\_available,

p.product\_id, p.product\_name, p.brand\_name,

pc.product\_category, pc.product\_subcategory, pc.product\_family,

s.store\_id, s.store\_name, s.store\_type, s.store\_city, s.store\_state, s.store\_country, s.coffee\_bar, s.salad\_bar,

c.customer\_id, c.state\_province, c.country, c.marital\_status,(CASE WHEN c.num\_children\_at\_home > 0 THEN 'YES' ELSE 'NO' END) AS has\_children, c.education, c.occupation, (CASE WHEN c.num\_cars\_owned > 0 THEN 'YES' ELSE 'NO' END) AS has\_car,

pr.media\_type,

t.time\_id, t.the\_date, t.the\_day, t.the\_month, t.month\_of\_year, t.quarter

FROM (SELECT \* FROM sales\_fact\_1998 UNION SELECT \* from sales\_fact\_dec\_1998) sf

JOIN product p ON p.product\_id = sf.product\_id

JOIN product\_class pc ON pc.product\_class\_id = p.product\_class\_id

JOIN store s ON s.store\_id = sf. store\_id

JOIN customer c ON c.customer\_id = sf.customer\_id

JOIN promotion pr ON pr.promotion\_id = sf.promotion\_id

JOIN time\_by\_day t ON t.time\_id = sf.time\_id

ORDER BY t.the\_date;

View to describe the database with relations

#--CREATE OR REPLACE VIEW foodmart\_dwh\_description AS

WITH p (tablename, description) AS

  (VALUES

('account', 'Internal accounts table'),

('agg\_c\_10\_sales\_fact\_1997',    'store sales facts with time details'),

('agg\_c\_14\_sales\_fact\_1997',    'Product, customer and promotion colreated store sales with  store and time details'),

('agg\_c\_special\_sales\_fact\_1997',   'redundant of agg\_c\_14\_sales\_fact\_1997'),

('agg\_g\_ms\_pcat\_sales\_fact\_1997',   'store wise sales and customercount by categories of customer and product, and by time'),

('agg\_lc\_06\_sales\_fact\_1997',   'Time based store sales fact by location'),

('agg\_lc\_100\_sales\_fact\_1997',  'Store sales correlated by customer and product with store location and time details'),

('agg\_ll\_01\_sales\_fact\_1997',   'store sales correlated by Prodcut, time and customer'),

('agg\_l\_03\_sales\_fact\_1997',    'store sales fact correlated by time and customer'),

('agg\_l\_04\_sales\_fact\_1997',    'time base store sales fact'),

('agg\_l\_05\_sales\_fact\_1997',    'store sales fact correlated by product, cutomer, promotion and store'),

('agg\_pl\_01\_sales\_fact\_1997',   'store sales fact correlated by product, customer and time'),

('category',    'expense category'),

('employee\_closure',    'employee closure details'),

('expense\_fact',    'expenses in appropriate currency, category and time'),

('inventory\_fact\_1997', 'inventory fact correlated by product, time, warehouse and store for 1997'),

('inventory\_fact\_1998', 'inventory fact correlated by product, time, warehouse, store for 1998'),

('position',    'Employee position details'),

('product', 'Product details'),

('product\_class',   'product category table'),

('promotion',   'customer promotion information table'),

('region',  'Region information table'),

('reserve\_employee',    'table with details of employees on reserve in corrlation with  position, store, supervisor and department'),

('currency',    'Currency conversion ratio by date'),

('customer',    'customer dedtails table'),

('days',    'week days table'),

('department',  'department table'),

('employee',    'Employee details table'),

('salary',  'Salary details correlated by employee, department'),

('sales\_fact\_1997', 'Main total sales fact table for 1997 correlated by product, time, store, customer and promotion'),

('sales\_fact\_1998', 'Main total sales fact table for 1998 correlated by product, time, store, customer and promotion'),

('sales\_fact\_dec\_1998', 'total sales fact table for 1998 in December month correlated by product, time, store, customer and promotion'),

('store',   'Store details table'),

('store\_ragged',    'Store details table'),

('time\_by\_day', 'table for time entry with id'),

('warehouse',   'Warehouse details table'),

('warehouse\_class', 'Warehouse Category table'))

SELECT t1.\*, p.description, t2.table\_columns FROM (

SELECT

    c1.TABLE\_NAME AS table\_name,

    c1.COLUMN\_NAME AS column\_name,

    (SELECT TABLE\_NAME

        FROM information\_schema.COLUMNS AS c2

        WHERE

            (CASE

                WHEN c2.TABLE\_NAME = 'time\_by\_day' THEN 'time\_id'

                ELSE

            CONCAT(c2.TABLE\_NAME, '\_id') END )= c1.COLUMN\_NAME

            AND c2.TABLE\_SCHEMA = 'foodmartdwh'

            AND c2.TABLE\_NAME <> c1.TABLE\_NAME

        LIMIT 1) AS related\_table

FROM information\_schema.COLUMNS AS c1

WHERE c1.TABLE\_SCHEMA = 'foodmartdwh') AS t1

JOIN (SELECT tb.TABLE\_NAME, GROUP\_CONCAT(COLUMN\_NAME) AS table\_columns FROM information\_schema.COLUMNS AS tb

        WHERE tb.TABLE\_SCHEMA = 'foodmartdwh'

        GROUP BY TABLE\_NAME) t2

    ON t2.TABLE\_NAME =  t1.TABLE\_NAME

JOIN p ON p.tablename = t1.table\_name

WHERE t1.related\_table IS NOT NULL;

: