



**Business School**  
UNIVERSITY OF COLORADO **DENVER**

**Information Systems Program**

# **Module 4**

## **Materialized View Processing and Design**

### **Lesson 3: Query Rewriting Rules**



# Lesson Objectives

- Explain query rewriting process
- Apply matching rules
- Reflect on complexity of query rewriting compared to query modification

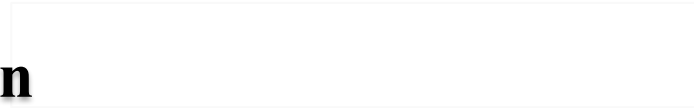


# Query Rewriting Overview

Substitution process  
matching materialized  
view and query



Replace fact and  
dimension tables in a  
query with a

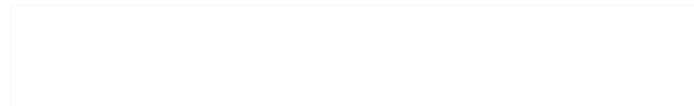


materialized view  
More complex than  
query modification  
process for traditional

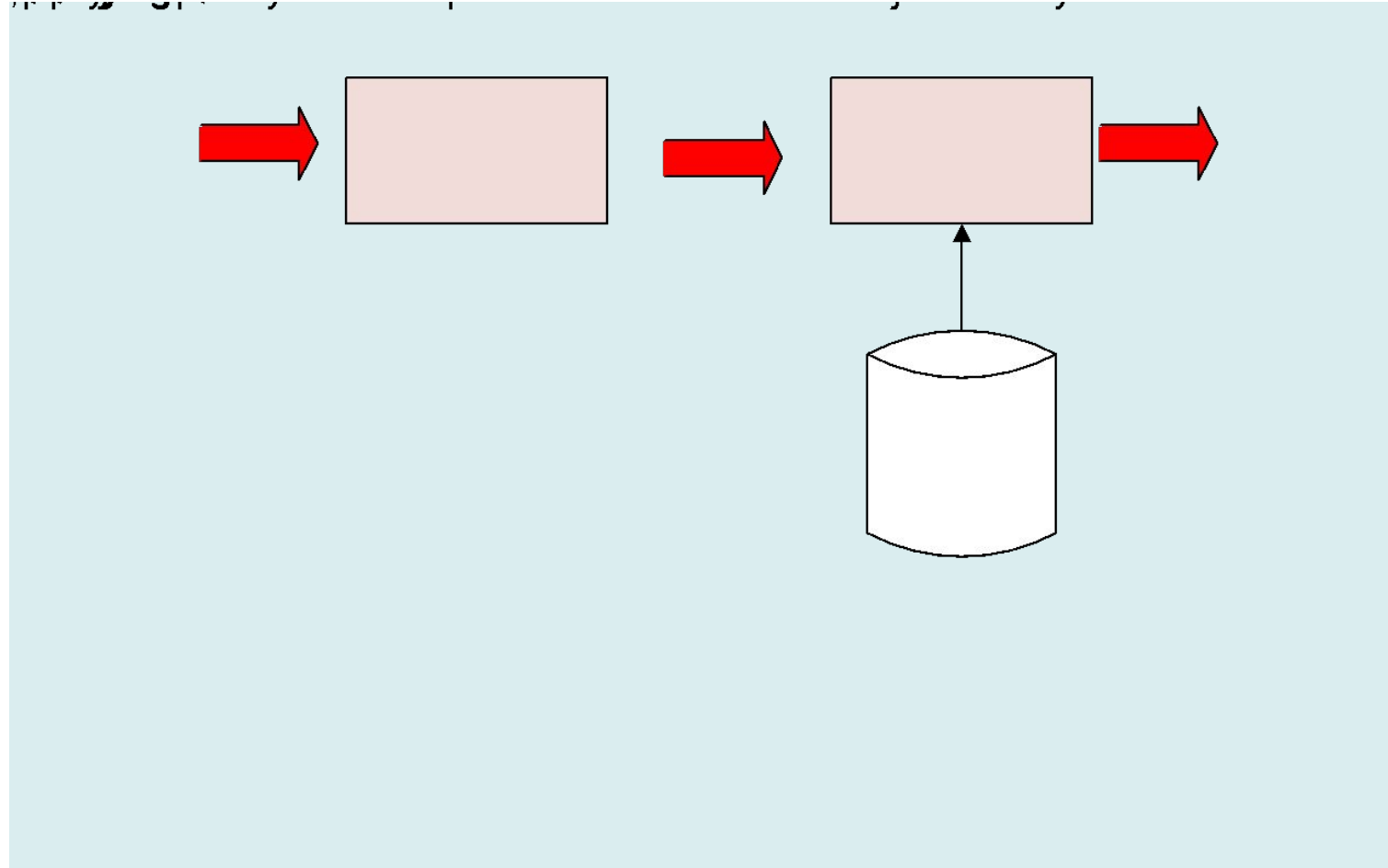


views

Evaluate performance  
improvements over the  
original query



# Query Rewriting Process



# Matching Rules

Row  
conditions

Grouping  
columns

Grouping  
dependencies

Aggregate  
functions

- MV rows contain query rows.

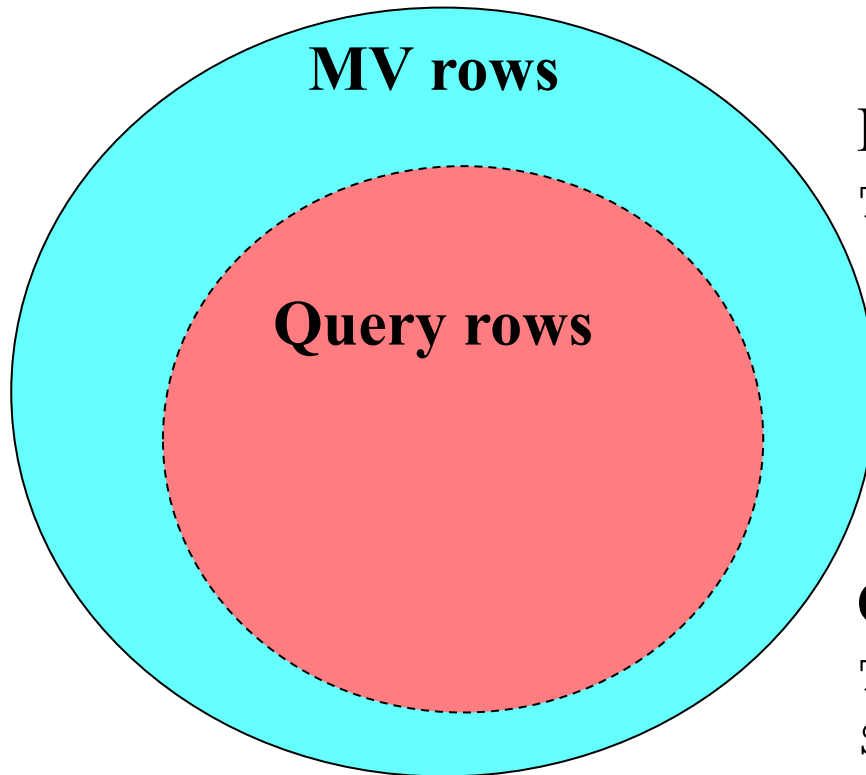
- Query conditions at least as grouping columns

- Query grouping match or derivable by FDs on aggregate functions

- Query aggregate functions match or derivable from MV aggregate functions.



# Matching for Rows



MV

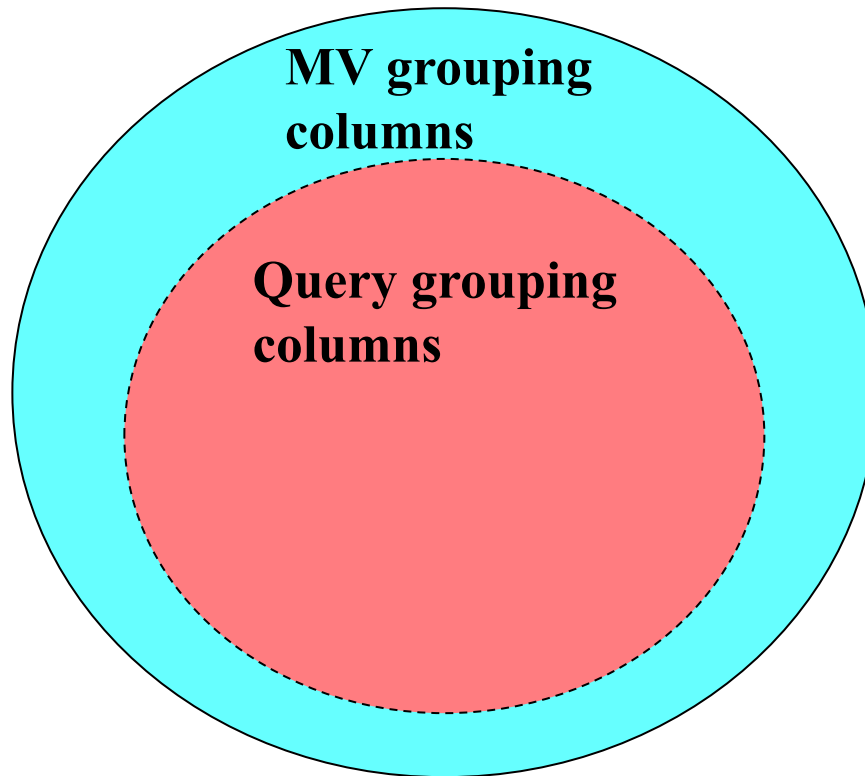
`TimeYear > 2014`

Query

`TimeYear = 2016 AND  
StoreNation = 'USA'`



# Matching for Grouping Columns



MV grouping:

- TimeYear
- StoreCity

Query grouping:

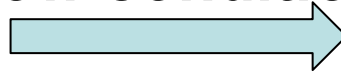
- StoreCity



# Combined Matching Example

## Row conditions

*MV*



*Query*

TimeYear > 2014 AND  
StoreNation IN ('USA', 'Canada')

TimeYear = 2016  
AND StoreNation = 'USA'

## Grouping columns

*MV*



*Query*

StoreState, TimeYear

StoreState

## Grouping dependencies

*MV*



*Query*

StoreId, TimeYear

StoreCity,

TimeYear

## Aggregate functions

*MV*



*Query*

SUM(DollarSales), COUNT(\*)

AVG(DollarSales)





# Summary

- Essential part of summary data management
- Matching rules for row conditions, grouping columns, grouping dependencies, and aggregate calculations
- Evaluation of rewritten queries by optimizing compiler

