sqlbi

sqlbi

sqlbi.com

# Vertipaq vs OLAP

## Change Your Data Modeling Approach

presented by
Marco Russo
marco.russo@sqlbi.com

sqlbi.com

sqlbi

## Who am I?

- marco.russo@sqlbi.com
- Independent consultant
- 15+ years on SQL Server
- 10+ years on BI & OLAP
- Book author
- Founder of SQLBI.COM

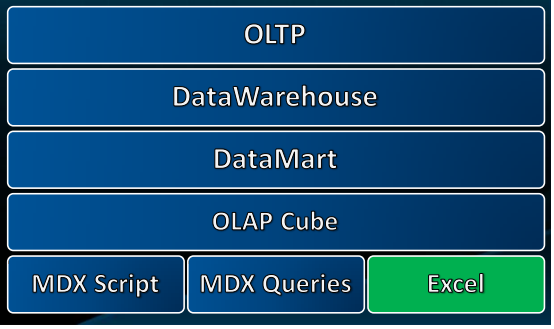

## Agenda

- OLAP Modeling (aka BISM Multidimensional)
- Vertipaq Modeling (aka BISM Tabular)
- Scenarios
- Thoughts

sqlbi

## Traditional OLAP with UDM

NOW CALLED BISM MULTIDIMENSIONAL




## OLAP vs Vertipaq

| OLAP<br>BISM Multidimensional | Vertipaq<br>BISM Tabular |
|---|---|
| • Dimensional Modeling<br>  • Facts, Dimensions<br>  • Complex Relationships<br>• MDX<br>  • Script<br>  • Powerful, complex | • Relational Modeling<br>  • Tables<br>  • Basic Relationships (1:N)<br>• DAX<br>  • Calculated Columns<br>  • Measures |

## Vertipaq

NOW CALLED BISM TABULAR

- Simpler data model
- Less dimensional tools
- Needs more memory
- Amazingly fast

- But…

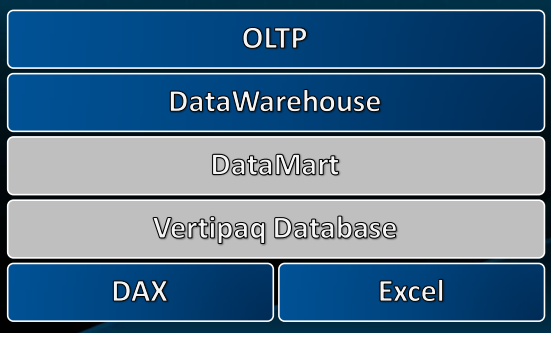
  Is speed the only advantage?

## Dimensional Modeling

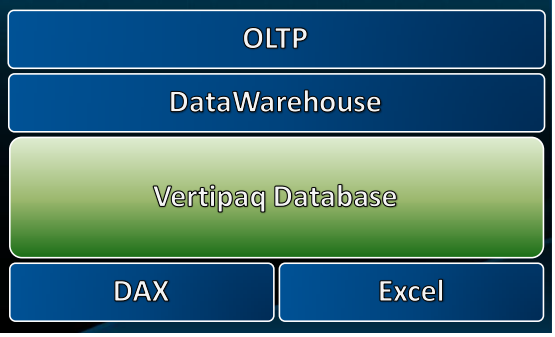
Why do we use dimensional modeling?

- Easiness of use?
  - Any DWH can turn into dimensional through views
- Speed!
  - No DWH model will ever beat a good DM
- Excel!
  - OLAP cubes need to be built on DM
  - Excel query OLAP cubes
- In 2011, Dimensional Modeling is the only solution

## BI Solutions with Vertipaq…




## BI Solutions with Vertipaq…



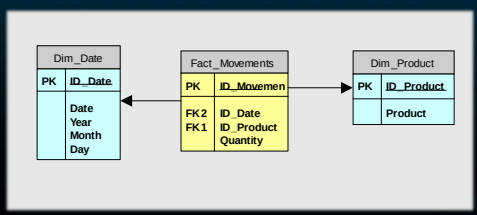

CALCULATE WAREHOUSE AVAILABILITY FROM TRANSACTIONS

## Scenario
## Warehouse Stock Analysis

sqlbi.com

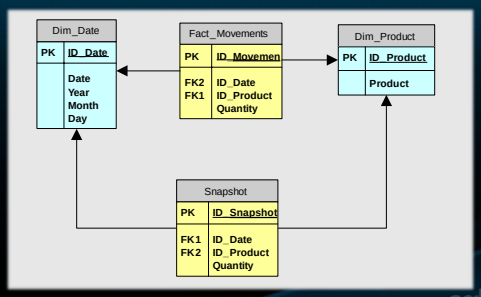
## Warehouse Stock Analysis

RELATIONAL MODEL FOR TRANSACTIONS



# Warehouse Stock Analysis

RELATIONAL MODEL FOR TRANSACTIONS




# Warehouse Stock Analysis

- Common solution:
  - Snapshot table with quantity on hold for each product/warehouse/day
  - ETL calculation made by summing all transactions up to a given day
- BISM Tabular
  - Can avoid snapshot by making up-to-date calculation at query time
  - Reduces memory, leverages on Vertipaq engine

# Warehouse Stock Analysis

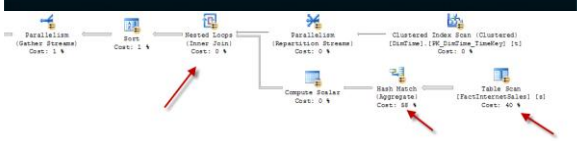
SQL QUERY TO GENERATE MONTHLY SNAPSHOT

```
SELECT
    t.TimeKey,
    t.FullDateAlternateKey,
    s1.*
FROM
    dbo.DimTime t
CROSS APPLY ( SELECT
                s.ProductKey,
                SUM(s.Quantity) AS Stock
              FROM
                dbo.FactMovements s
              WHERE
                s.OrderDateKey <= t.TimeKey
              GROUP BY
                s.ProductKey ) s1
WHERE
    DayNumberOfMonth = 1
```

# Warehouse Stock Analysis

SQL QUERY PLAN TO GENERATE MONTHLY SNAPSHOT

- For each date, a Full Scan of the fact table is required in order to generate the snapshot



# Warehouse Stock Analysis

MDX QUERY WITHOUT A SNAPSHOT

- The MDX query can be slow:

```
WITH MEMBER MEASURES.Stock AS
    SUM( NULL : [Date Order].[Calendar].CURRENTMEMBER,
         [Measures].[Quantity] )
SELECT
    Stock ON 0,
    NON EMPTY [Product].[Product].[Product].members
            * [Date Order].[Calendar].[Month].MEMBERS ON 1
FROM [Movements]
```

# Warehouse Stock Analysis

DAX QUERY DOES NOT REQUIRE SNAPSHOT

- The DAX query is faster:

```
Stock :=
    CALCULATE(
        SUM( FactMovements[OrderQuantity] ),
        FILTER(
            ALL( DimTime ),
            DimTime[TimeKey] <= MAX( DimTime[TimeKey] )
        )
    )
```

COUNT NEW AND LOST CUSTOMERS

# Scenario
# Customer Analysis

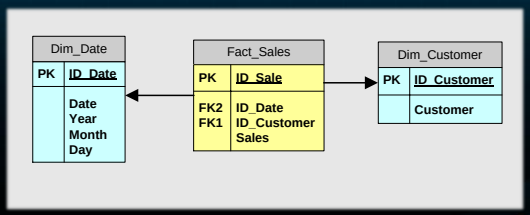
sqlbi.com

## Customer Analysis

- The question:
  - How many new distinct customers we had this month?
- Dimensional model
  - Complex and slow query (either SQL or MDX)
    - Slowness is caused by looking for the lack of an event in a dimensional model
  - Optimization: monthly snapshot of new customers
    - Possible shortcut: saving date of first and last sale for each customer
    - Extraction logic embedded in ETL – not flexible

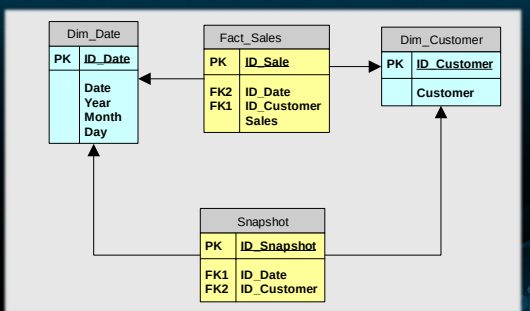
## Customer Analysis

INITIAL DATA MODEL IS A CLASSICAL STAR SCHEMA



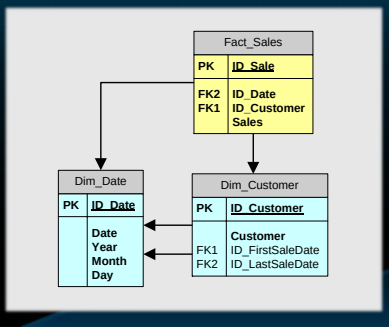

## Customer Analysis

SOLUTION BASED ON A SNAPSHOT – ETL REQUIRED




## Customer Analysis

SOLUTION WITH FIRST/LAST TRANSACTION DATE – UPDATE ON DIMENSION REQUIRED




## Customer Analysis

MDX QUERY RELIES ON DISTINCT COUNT MEASURE

```
WITH
MEMBER MEASURES.TotalCustomers AS
    AGGREGATE( NULL : [Date].[Calendar].CURRENTMEMBER,
               [Measures].[Customer Count])
MEMBER MEASURES.NewCustomers AS
    MEASURES.TotalCustomers
    - AGGREGATE( NULL : [Date].[Calendar].PREVMEMBER,
                 [Measures].[Customer Count])
MEMBER MEASURES.ReturningCustomers AS
    MEASURES.[Customer Count] - MEASURES.NewCustomers
SELECT
  { [Measures].[NewCustomers],
    [Measures].[Customer Count],
    [Measures].[ReturningCustomers] } ON COLUMNS,
  [Date].[Calendar].[Month].MEMBERS ON ROWS
FROM [Adventure Works]
```

## Customer Analysis

WHY DISTINCT COUNT MEASURES ARE EXPENSIVE IN OLAP

- Distinct Count calculation in OLAP is expensive
  - Separate measure group
    - Adding a distinct count measure requires modifications in OLAP schema
  - Expensive processing (ORDER BY required)
  - Different partitioning for query optimization

## Customer Analysis

- BISM Tabular
  - Query in DAX is fast enough
    - Good to get number and also to extract customers list
  - Extraction logic in DAX, not in ETL
    - Easy to change
  - No special ETL and data modeling required
    - Just because distinct count is fast

## Customer Analysis

DAX QUERY HEAVILY USE DISTINCTCOUNT WITHOUT PERFORMANCE ISSUES

```
NewCustomers :=
CALCULATE(
    DISTINCTCOUNT( FactInternetSales[CustomerKey] ),
    FILTER( ALL( DimTime ),
            DimTime[TimeKey] <= MAX( DimTime[TimeKey] )
    )
)
- CALCULATE(
    DISTINCTCOUNT( FactInternetSales[CustomerKey] ),
    FILTER( ALL( DimTime ),
            DimTime[TimeKey] < MIN( DimTime[TimeKey] )
    )
  )
```

TRANSITION OF AN ATTRIBUTE OVER TIME IN SCD2

## Scenario
## Transition Matrix

sqlbi.com

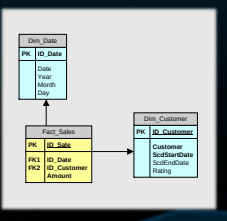
## Scenario: Transition Matrix

- Examine transition of an attribute over time in a SCD2 Customer Dimension
- The real question:
  - How many customers moved from rating AAA to rating AAB in the last month?
- Complex SQL query
- Complex OLAP model
  - Duplicate facts table in the model (as bridge table)

## Transition Matrix - OLAP

RELATIONAL MODEL SCD2

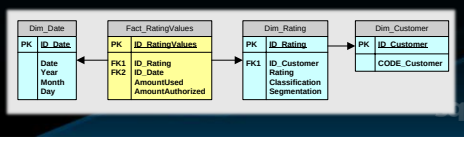
- Kimball SCD2 Dimension tracks changes in Rating for a customer



# Transition Matrix - OLAP

RELATIONAL MODEL FOR MULTIDIMENSIONAL

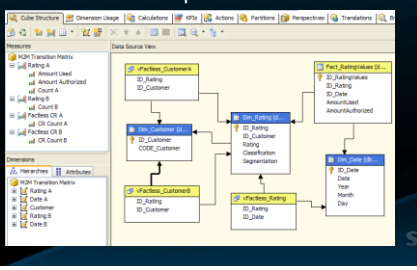
- Relational model needs translation
  - Dim_Rating is a SCD Type II on customer ratings
  - Dim_Customer is a SCD Type I
  - Fact_RatingValues is a snapshot of ratings measured monthly
    - In addition to the rating, it contains measures on used and authorized credit amount on the corresponding period




# Transition Matrix - OLAP

DATA SOURCE VIEW FOR ANALYSIS SERVICES

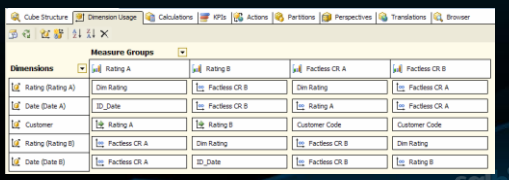
- More entities required to support many-to-many dimension relationships



# Transition Matrix - OLAP

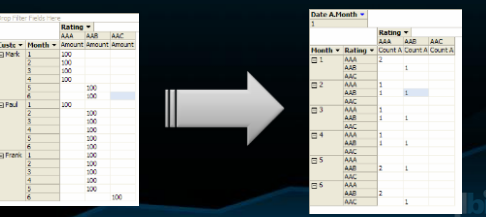
DIMENSION USAGE IN ANALYSIS SERVICES

- Two measure groups and two dimensions for Rating A and Rating B
- Complete relationships between all dimensions and measure groups to get maximum navigation flexibility




# Transition Matrix - OLAP

- Rating changes
  - E.g.: starting from January rating (see filter) on columns, see resulting rating in following months on rows



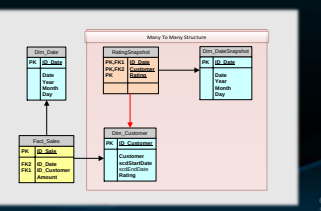
# Scenario: Transition Matrix

- BISM Tabular
  - Simpler data model (facts table are not duplicated)
  - Complexity in DAX measures
  - Easier to adapt to existing relational models
    - Calculated columns to avoid snapshot tables

# Transition Matrix – Tabular

RELATIONAL MODEL FOR MULTIDIMENSIONAL

- No translation required
  - Dim_Customer is still a SCD Type II
  - Rating_Snapshot is a snapshot of ratings measured monthly



## Transition Matrix - Tabular

NO SPECIAL DATA MODELING, JUST A DAX MEASURE

- NumOfCustomers :=

```
CALCULATE (
    COUNTROWS (DISTINCT (Dim_Customers[Customer])),
    FILTER (
        Dim_Customers,
        CALCULATE (
            COUNTROWS (RatingSnapshot),
            RatingSnapshot[CustomerSnapshot]
            = EARLIER (Dim_Customers[Customer])
        ) > 0
        && (Dim_Customers[scdStartDate]
            <= MAX (Dim_Date[ID_Date]) )
        && (
            Dim_Customers[scdEndDate]
                > MIN (Dim_Date[ID_Date])
            || ISBLANK (Dim_Customers[scdEndDate])
        )
    )
)
```

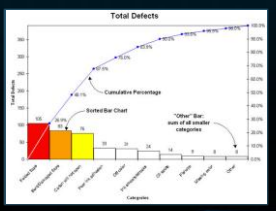
FROM TWO LOOPS TO SINGLE CALCULATION

## Scenario
## ABC Classification

sqlbi.com

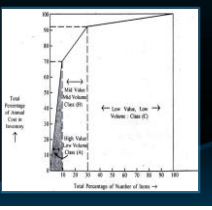
## ABC and Pareto Analysis

- Pareto principle
  - 80% of effects come from 20% of the causes



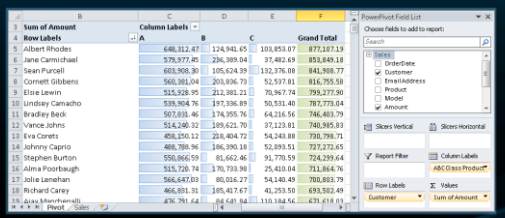

## ABC and Pareto Analysis

- ABC Analysis
  - Class A contains items for 70% of total value
  - Class B contains items for 20% of total value
  - Class C contains items for 10% of total value



## ABC Analysis Usage

- The ABC class can be used to slice data




## ABC Analysis with Excel

The formula is straightforward:

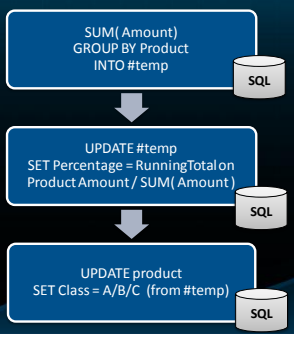
=IF(H6<=0,7;"A";IF(H6<=0,9;"B";"C"))

| Product | TotalSold | Incremental | GrandTotal | Incr.% | Class |
|---|---|---|---|---|---|
| Mountain-200 Black, 38 | 4.400.592,80 | 4.400.592,80 | 109.846.381,40 | 4,01% | A |
| Mountain-200 Black, 42 | 4.009.494,76 | 8.410.087,56 | 109.846.381,40 | 7,66% | A |
| Mountain-200 Silver, 38 | 3.693.678,03 | 12.103.765,59 | 109.846.381,40 | 11,02% | A |
| Mountain-200 Silver, 42 | 3.438.478,86 | 15.542.244,45 | 109.846.381,40 | 14,15% | A |
| Mountain-200 Silver, 46 | 3.434.256,94 | 18.976.501,39 | 109.846.381,40 | 17,28% | A |
| Mountain-200 Black, 46 | 3.309.673,22 | 22.286.174,61 | 109.846.381,40 | 20,29% | A |
| Road-250 Black, 44 | 2.516.857,31 | 24.803.031,92 | 109.846.381,40 | 22,58% | A |
| Road-250 Black, 48 | 2.347.655,95 | 27.150.687,88 | 109.846.381,40 | 24,72% | A |
| Road-250 Black, 52 | 2.012.447,78 | 29.163.135,65 | 109.846.381,40 | 26,55% | A |
| Road-150 Red, 56 | 1.847.818,63 | 31.010.954,28 | 109.846.381,40 | 28,23% | A |
| ML Mountain Frame - Black, 42 | 901.590,24 | 76.101.720,96 | 109.846.381,40 | 69,28% | A |
| Road-650 Red, 62 | 879.827,94 | 76.981.548,90 | 109.846.381,40 | 70,08% | B |
| Road-650 Red, 48 | 878.666,66 | 77.860.215,56 | 109.846.381,40 | 70,88% | B |
| Road-650 Black, 58 | 869.632,78 | 78.729.848,34 | 109.846.381,40 | 71,67% | B |
| Touring-2000 Blue, 54 | 772.302,01 | 79.502.150,36 | 109.846.381,40 | 72,38% | B |
| Touring-1000 Blue, 54 | 743.353,03 | 80.245.503,38 | 109.846.381,40 | 73,05% | B |
| Road-550-W Yellow, 44 | 717.825,91 | 80.963.329,29 | 109.846.381,40 | 73,71% | B |
| Road-350-W Yellow, 44 | 694.003,92 | 81.657.333,21 | 109.846.381,40 | 74,34% | B |
| Touring-1000 Yellow, 54 | 667.158,15 | 82.324.491,36 | 109.846.381,40 | 74,95% | B |
| Touring-2000 Blue, 60 | 635.723,72 | 82.960.215,08 | 109.846.381,40 | 75,52% | B |
| Road-450 Red, 52 | 621.103,74 | 83.581.318,82 | 109.846.381,40 | 76,09% | B |

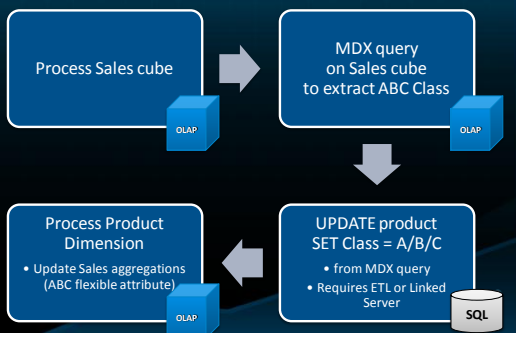
## Scenario: ABC Classification

- Classical implementation:
  - Heavy Calculation in ETL
  - Could be very long in SQL
  - Faster in MDX
    - But requires double cube processing in order to store ABC classification result in a dimension attribute

## ABC in SQL




## ABC in MDX




## Scenario: ABC Classification

- BISM Tabular
  - Based on calculated columns
  - Calculation done at processing time
    - Good performance, leverages on Vertipaq
  - No need for double processing

## ABC Analysis

- Tables: Product, SalesOrderDetail

```
SalesAmountProduct =
    CALCULATE( SUM( SalesOrderDetail[LineTotal] ) )
CumulatedProduct =
    SUMX(
        FILTER( Product,
                Product[SalesAmountProduct]
                    >= EARLIER( Product[SalesAmountProduct] ) ),
        Product[SalesAmountProduct] )
```

## ABC Analysis

- Calculate the class

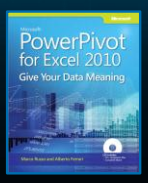
```
SortedWeightProduct =
    Product[CumulatedProduct]
        / SUM( Product[SalesAmountProduct] )

[ABC Class Product] =
    IF( Product[SortedWeightProduct] <= 0.7, "A",
        IF( Product[SortedWeightProduct] <= 0.9, "B", "C" ) )
```

## Data Modeling Considerations

- We are used to put everything in a dimensional model
  - Even when we haven't one (like snapshot)
  - Vertipaq is a game changer for problems that don't fit into a dimensional model

## Call to action

- Consider alternative models for Data Marts consumed by Vertipaq
- Learn DAX
- Consider BISM Tabular working side-by-side with BISM Multidimensional
  - Specific models can be better solved by BISM Tabular
  - When possible, adapt the technology to the Data Model, not the data model to the technology




A BOOK TO LEARN DAX NOW!

**PowerPivot for Excel 2010: give your data meaning**

## Questions and Answers

## Links

- SQLBI Website
  **www.sqlbi.com**
- PowerPivot Workshop
  **www.powerpivotworkshop.com**
- Marco Russo blog
  **www.sqlblog.com/blogs/marco_russo**
- Alberto Ferrari blog
  **www.sqlblog.com/blogs/alberto_ferrari**

For any question contact us at
**info@sqlbi.com**

## Thank you!

## Coming up…

| Speaker | Title | Room |
|---|---|---|
| Klaus Aschenbrenner | Understanding SQL Server Execution Plans | Aintree |
| Thomas Kejser | Finding the Limits | Lancaster |
| Alberto Ferrari | Many-to-Many Relationships in DAX | Pearce |
| Mark Whitehorn | MDX and DAX-compare and contrast | Boardroom |
| Bob Duffy | SQL tuning from the dot.net perspective | Empire |
| Francesco Quaratino | The forgotten DBA daily essential checklist | Derby |