



Smarter Trading: Harnessing the value of an Enterprise Data Warehouse

Abhishek Iyer, Data Warehouse Specialist,
IBM India Software Labs

InformationOnDemandIndia2011

The Premier Conference for Information Management
Manage. Analyze. Govern.

February 2, 2011

Hyatt Regency | Mumbai, India

Customer Experience: Bombay Stock Exchange (BSE), India



- Oldest stock exchange in Asia
- Established in 1875, it has a rich heritage spanning the 133 years of existence.
- First stock exchange in India to receive permanent recognition - in 1956 - from the Government of India under the Securities Contracts (Regulation) Act 1956
- Biggest stock exchange in India in terms of number of listed companies and market capitalization
- Almost every leading corporate in India has secured BSE's services in capital raising and is listed in BSE
- As a brand, BSE has been and is synonymous with the capital market in India. Its SENSEX is the benchmark equity index that reflects the health of the Indian economy.

2





Business Need

- Existing ***siloed*** approach was causing significant delays in knowledge sharing across departments
- To comply with regulations, BSE needed to provide certain reports on a regular basis to the Securities and Exchange Board of India (SEBI), a regulatory body for stock exchanges
- BSE was struggling to consolidate all of its customer data and generate the required reports in a timely manner
- BSE's previous two attempts of consolidation using Oracle and Sybase were not successful



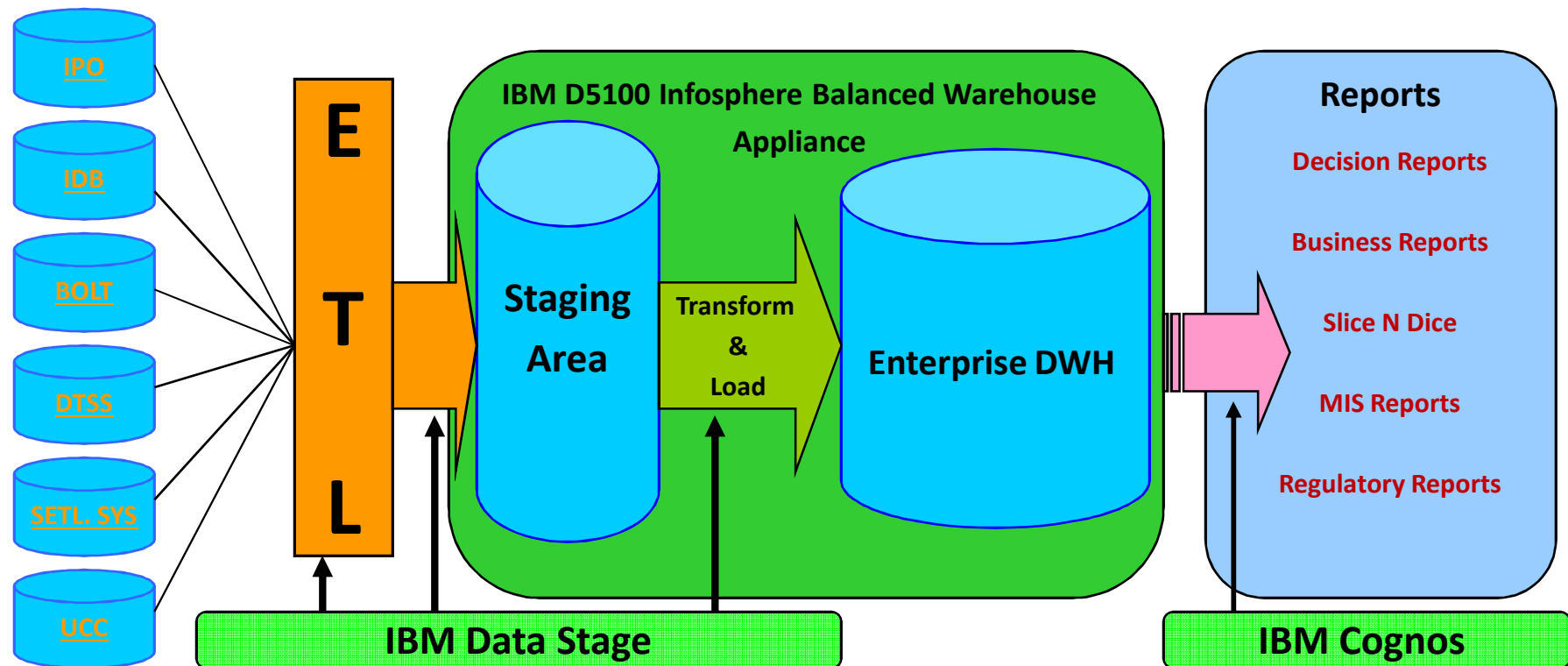


Objective

- BSE wanted to adopt new technology so it could provide more robust self-service capabilities to its customers.
- Wanted an enterprise data warehousing solution that would help overcome the major challenges:
 - Fastest implementation.
 - Lowest total cost of ownership.
 - Lowest implementation efforts.
 - Least involvement of System and Database Administrators.
 - Fast complex query execution time.



BSE Enterprise Data Warehouse Architecture



IBM Data Warehousing: Simplicity, Flexibility, Choice



Appliances



IBM Netezza

Flexible Integrated Systems



IBM Smart Analytics System

Custom Software, Hardware & Services



IBM InfoSphere Warehouse
IBM Servers / Storage
IBM Services

Warehouse Accelerators

Information Management Entry Points

(Information Server, MDM, Streams, etc)

Simplicity

The right mix of simplicity and flexibility

Flexibility

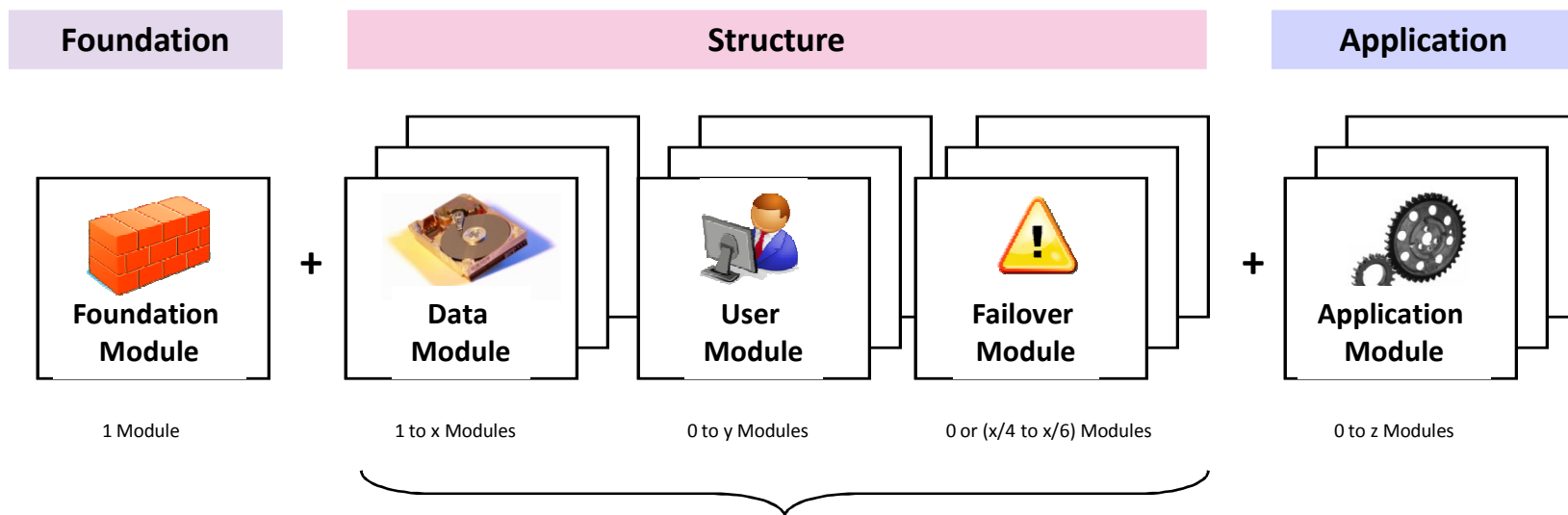




IBM InfoSphere Balanced Warehouse™ Appliance

Flexible modular architecture – How it Works

- Choose the way that your data warehouse solution develops
- Simply start with any foundation and just add modules as you require



Step 1:

Start with a single Foundation Module, the starting common foundation for each Balanced Warehouse (Note: at least 1 Data Module is required as well for a Balanced Warehouse)

Step 2:

If and when you need additional data handling capacity, number of users or failover functionality, add additional nodes from this group as needed. Note that 1 Data Module is needed for the minimum Balanced Warehouse configuration

Step 3:

Once the applicable structural modules are in place, application modules may be added to act as a dedicated application access point to your balanced warehouse



BSE EDW Solution specifics



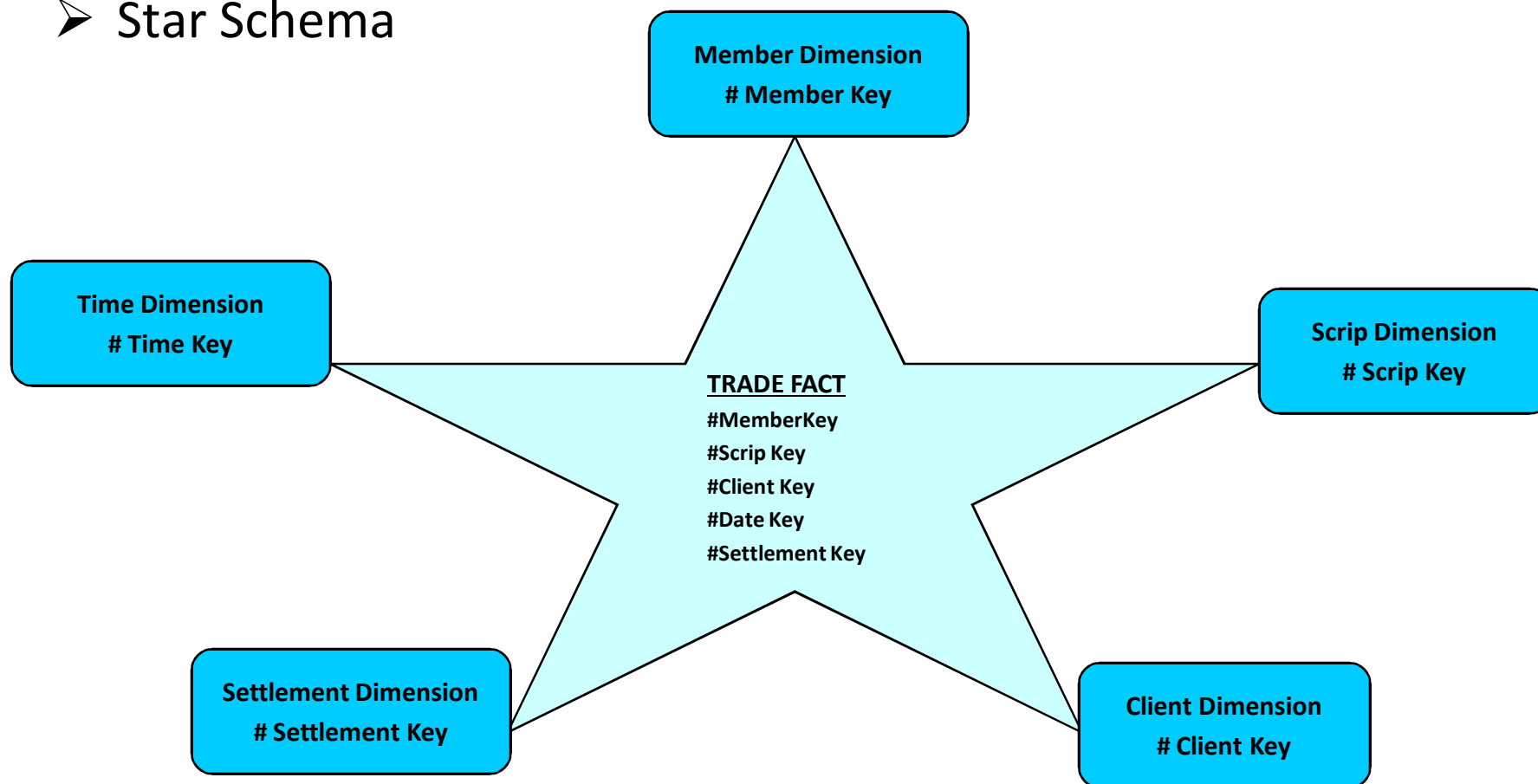
- Dimensional Data Model
 - Dimensions
 - Member, Scrip, Unique Client, Time, Settlement
 - Facts
 - Trades, Orders



BSE EDW Solution specifics



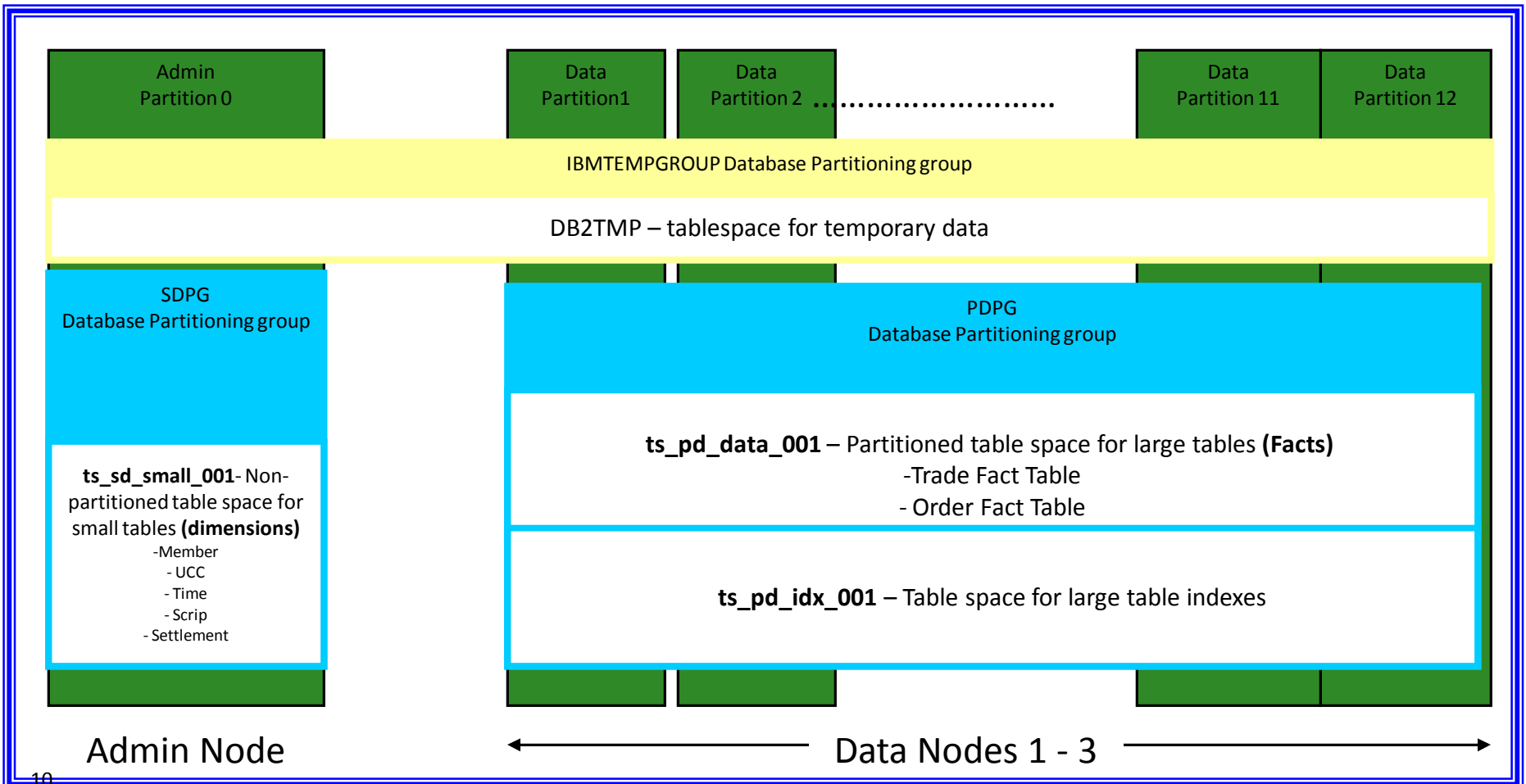
➤ Star Schema





Implementation based on IBM Best Practices

- Database objects placement
- ## One database instance



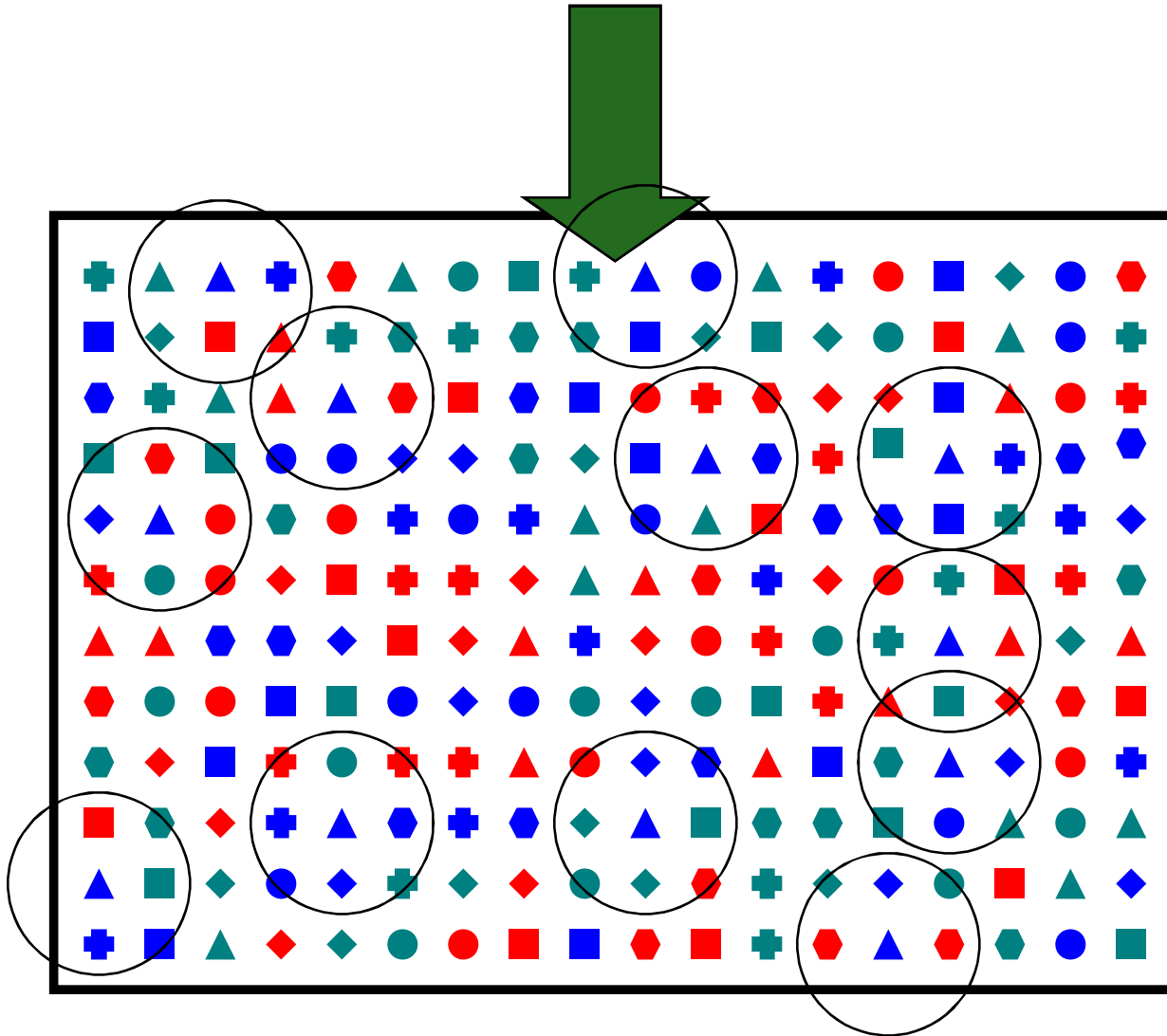
Implementation based on IBM Best Practices



- HASH Partitioning of Fact Table.
- RANGE Partitioning Of Fact Table.
- Multi Dimension Clustering Of Fact Table.
- Replicated MQT for DIMENSION tables to ensure Collocated Joins.

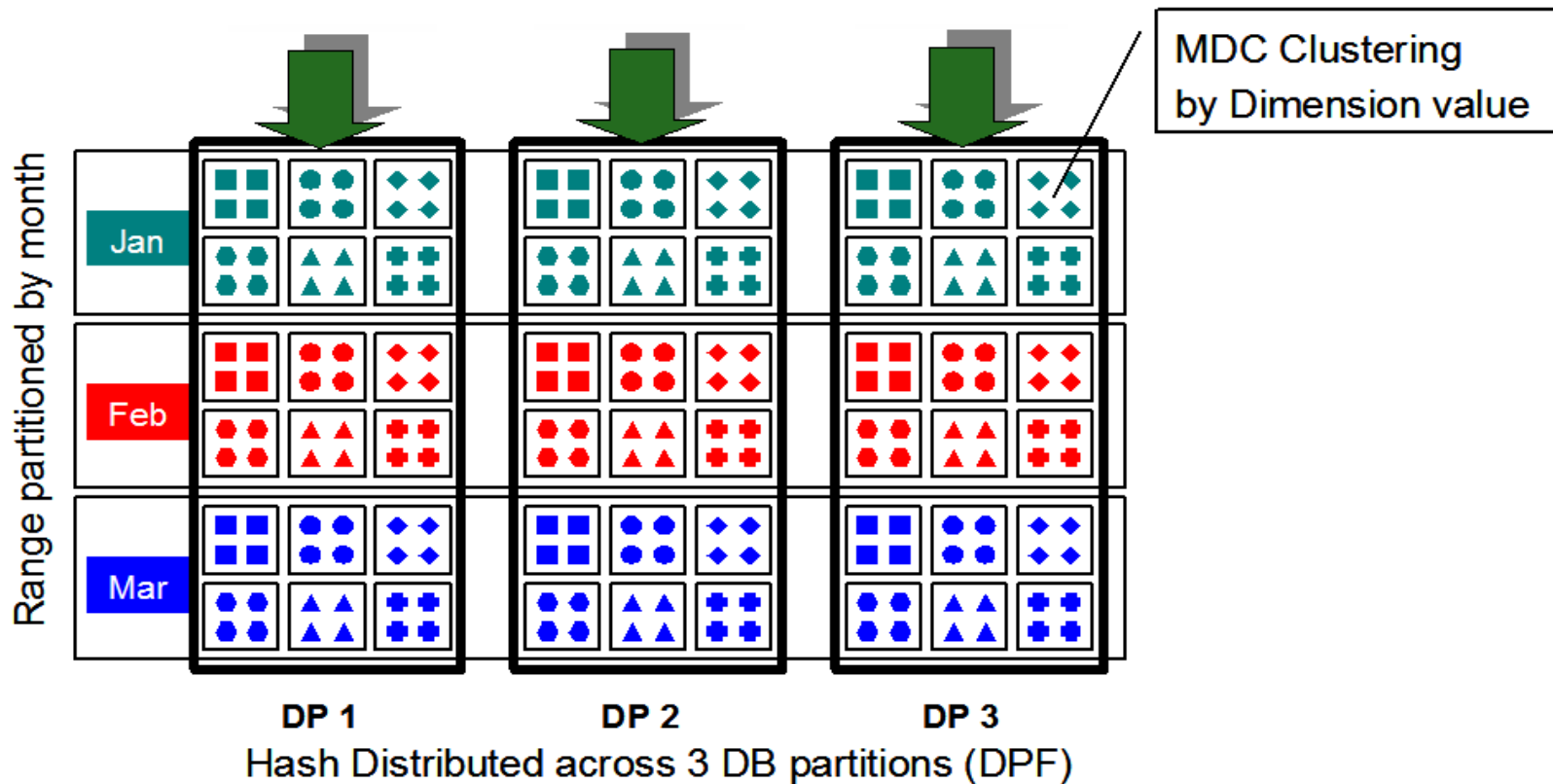


Data in One Monolithic Table





Hash distributed, Range partitioned & Clustered (MDC)





Major Performance gain claimed by BSE

- COMPLEX QUERY EXECUTION TIME
 - Reduced from **4 Hours** to **20 Minutes**
- SIMPLE QUERY EXECUTION TIME
 - Response Time in **Seconds**
- TIMELY DELIVERY OF REPORTS



Cost Benefit claimed by BSE

- Lowest Total Cost Of Ownership for 5 years.
- 5 TB Enterprise Data Warehouse implemented in just 7 months.
- 2 TB Storage space saving due to excellent DB2 Compression ratios.





Thank You

InformationOnDemandIndia2011

The Premier Conference for Information Management
Manage. Analyze. Govern.

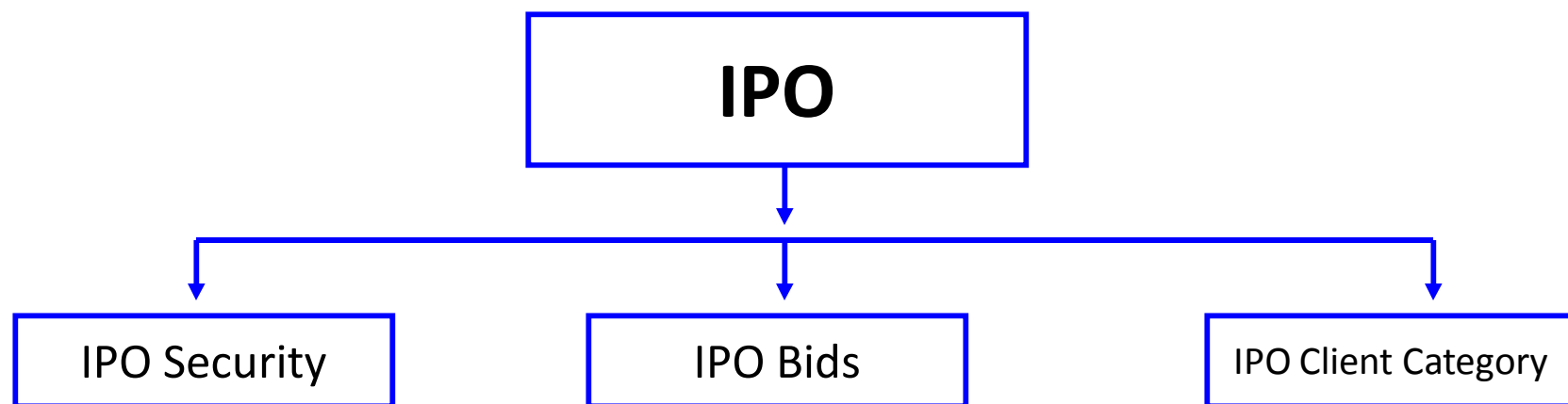
February 2, 2011

Hyatt Regency | Mumbai, India

BSE Enterprise Data Warehouse Architecture



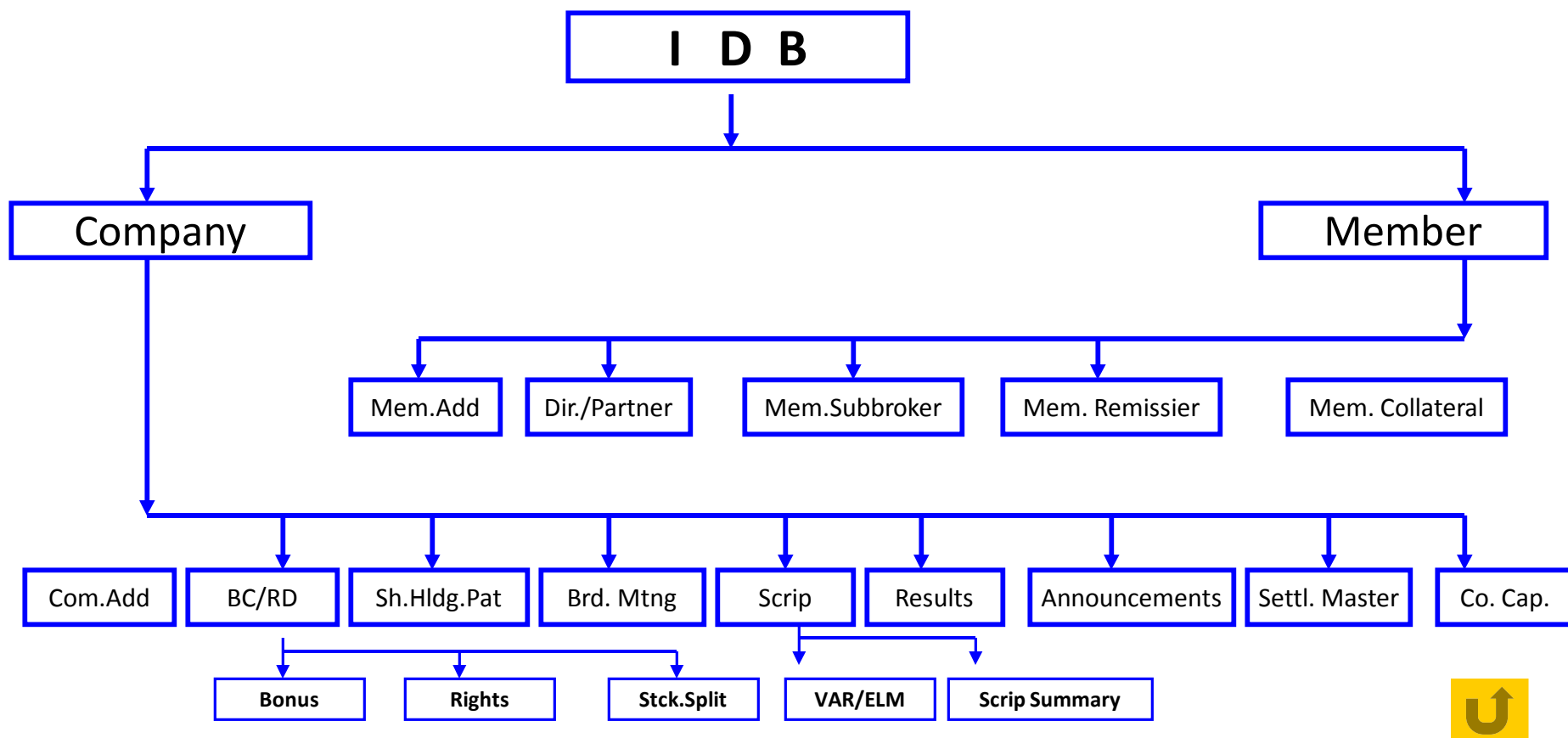
- Initial Public Offer



BSE Enterprise Data Warehouse Architecture



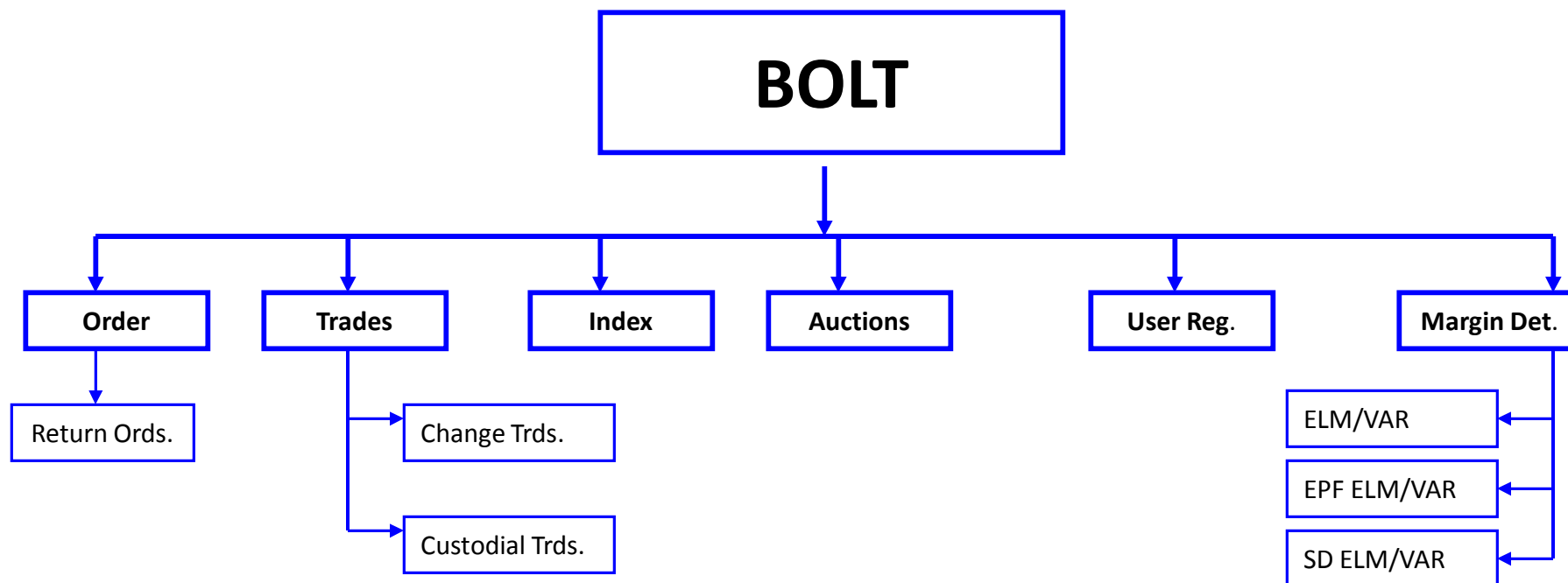
- Integrated Database (IDB)



BSE Enterprise Data Warehouse Architecture



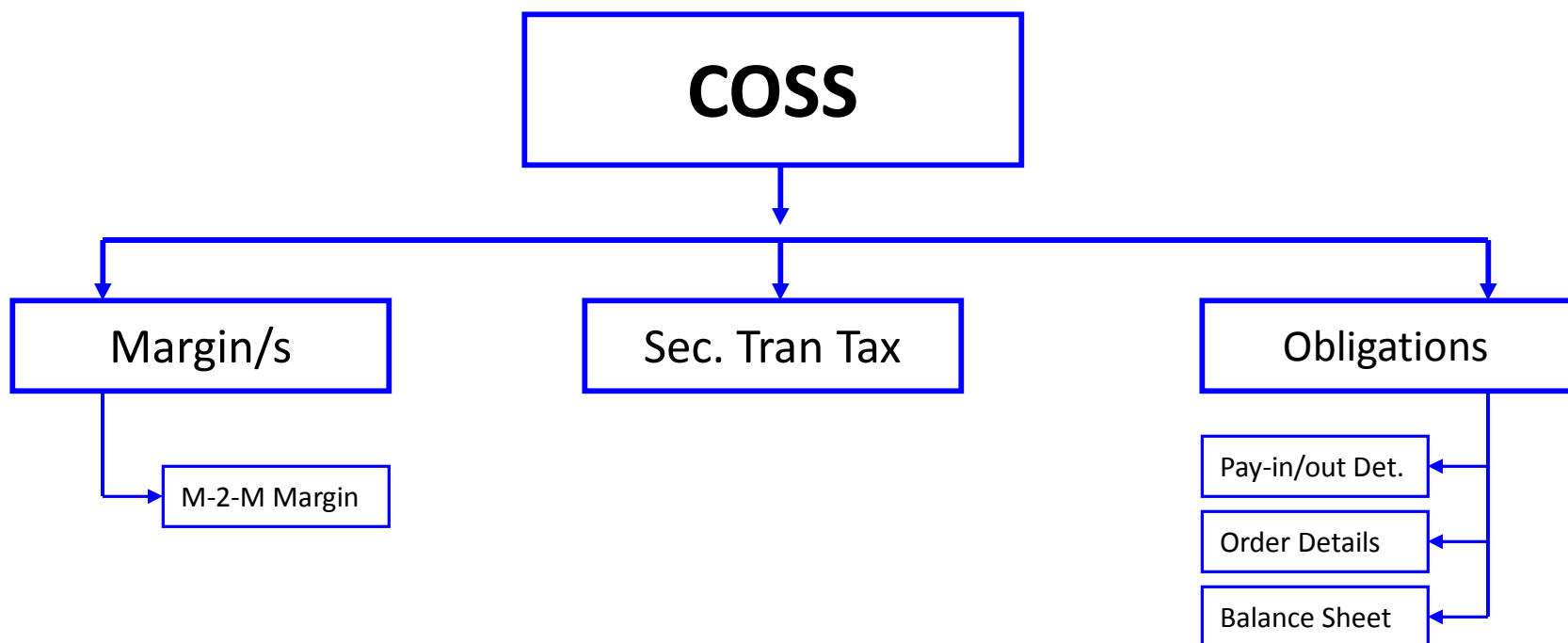
- BSE Online Trading System



BSE Enterprise Data Warehouse Architecture



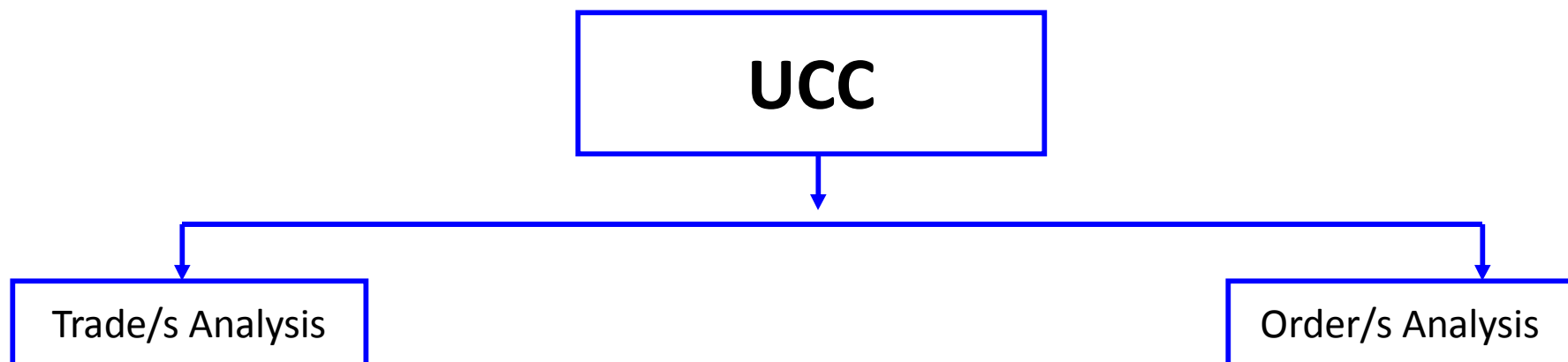
- Settlement (COSS)



BSE Enterprise Data Warehouse Architecture



- Unique Client Code (UCC)





Automatic data distribution using hashing & distribution map

```
CREATE TABLE customer (  
  cust_id VARCHAR(80)  
  ,gender CHAR(5)  
  DISTRIBUTE BY(cust_id);
```

```
CREATE TABLE sales (  
  cust_id VARCHAR(80)  
  ,qty INTEGER)  
  DISTRIBUTE BY(cust_id);
```

Customer	
cust_id	Gender
Smith	F
Bill	M
Woe	M
Zool	M
Mary	F

Sales	
cust_id	Qty
Smith	1
Smith	2
Smith	3
Zool	1
Zool	2
...	...

Hash(cust_id)

1	2	3	4	5	6	7	8	...	4096
0	1	2	0	1	2	0	1	2	0

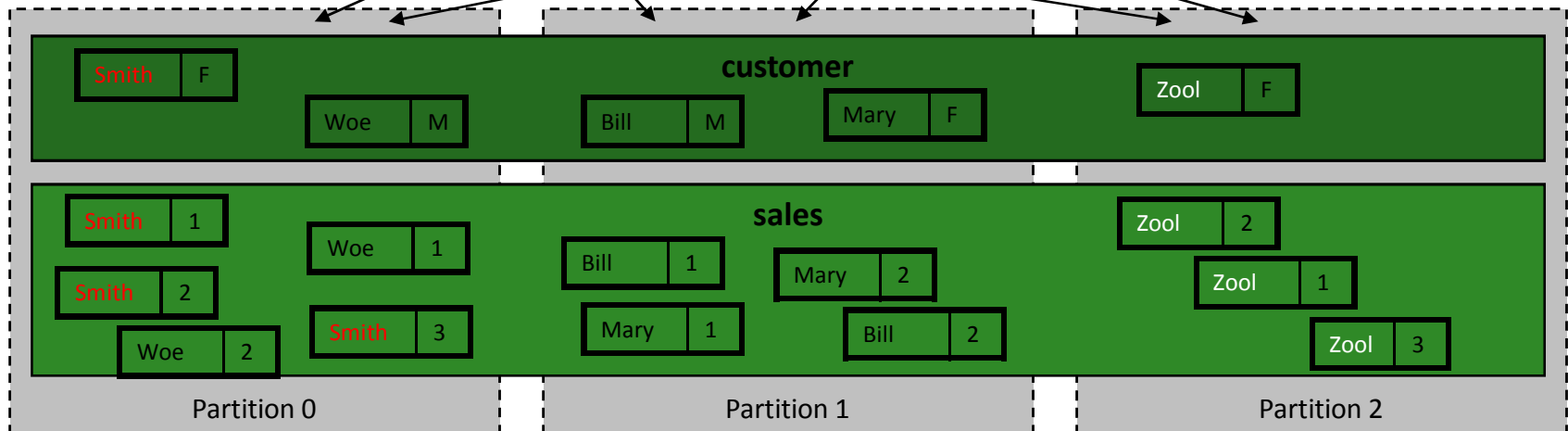
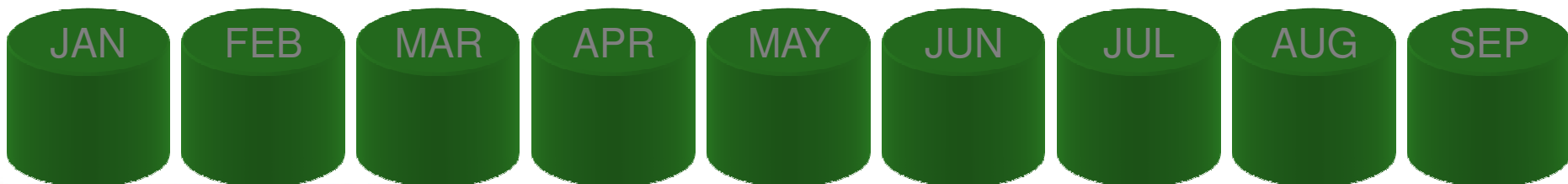




Table Partitioning by RANGE

- Allows you to create a table where *each range of the data in the table is stored separately*.
- For example, if you partition a table by month, all data for a given month will be kept together.
- In fact, internally, the database represents each range as a separate table and ranges can be scanned independently.
- Very quick and easy to attach or detach ranges of data from a partitioned table. Extremely useful for rolling in or out batches of data.

```
CREATE TABLE sales(sale_date DATE, customer INT, ...)
PARTITION BY RANGE(sale_date)
(STARTING '1/1/2011' ENDING '1/31/2011',
 STARTING '2/1/2011' ENDING '2/28/2011',
 STARTING '3/1/2011' ENDING '3/31/2011',
 STARTING '4/1/2011' ENDING '4/30/2011');
```





Multi Dimensional Clustering

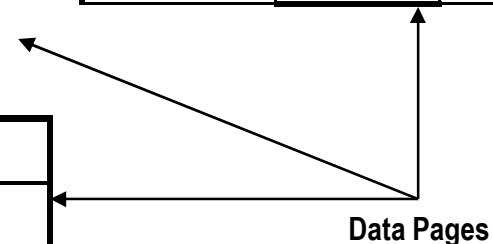
- A technique of clustering data along more than one dimension at the same time (based on value).
- It basically co-locates rows with the same key values in the same data page.

```
CREATE TABLE MDC_FACT ( SKU INT,  
                        store INT,  
                        sdate DATE,  
                        qty INT,  
                        amt DECIMAL (5,2))  
ORGANIZE BY DIMENSIONS (SKU, store, sdate)
```

SKU	Store	Date	Qty	Amt
101	21	04/02	1	1.50
101	21	04/02	1	1.50
101	21	04/02	3	4.10

SKU	Store	Date	Qty	Amt
101	7	04/02	2	3.00
101	7	04/02	1	1.50

SKU	Store	Date	Qty	Amt
101	7	04/01	6	8.11
101	7	04/01	2	3.00





Replicated MQTs to ensure collocated joins

- Small dimension tables can be replicated across all partitions using MQTs (materialized query tables).
- Very useful for enforcing collocation for small tables and small answer sets.

```
CREATE TABLE R_EMPLOYEE AS  
( SELECT EMPNO, FIRSTNAME, MIDINIT, LASTNAME, WORKDEPT FROM EMPLOYEE )  
DATA INITIALLY DEFERRED  
REFRESH IMMEDIATE  
IN REGIONTABLESPACE  
REPLICATED;
```

