IIT School of Applied Technology

ILLINOIS INSTITUTE OF TECHNOLOGY

information technology & management

526 Data Warehousing

February 10, 2016

Week 4 Presentation

Week 04 Topic: Dimensional Modeling: Basic Dimension Tables Techniques

- We will cover
  - Surrogate Key Implementation Considerations
  - Date Dimension Considerations
  - Roll-Playing Dimensions
  - Avoiding Too Many Dimensions
  - Junk Dimensions

Dimension Tables Revisited

- Contain descriptive attributes that are typically textual fields
- Shallow and wide
- Corresponds to entities that business interacts with
  - Customer, Employee, Products, Accounts
- Single column PK (typically a surrogate key) with a single column natural key

Product Dimension
Product Key (PK)
SKU Number (Natural Key)
Product Description
Brand Name
Category Name
Department Name
Package Type
Package Size
Abrasive Indicator
Weight
Weight Unit of Measure
Storage Type
Shelf Life Type
Shelf Width
Shelf Height
Shelf Depth
...

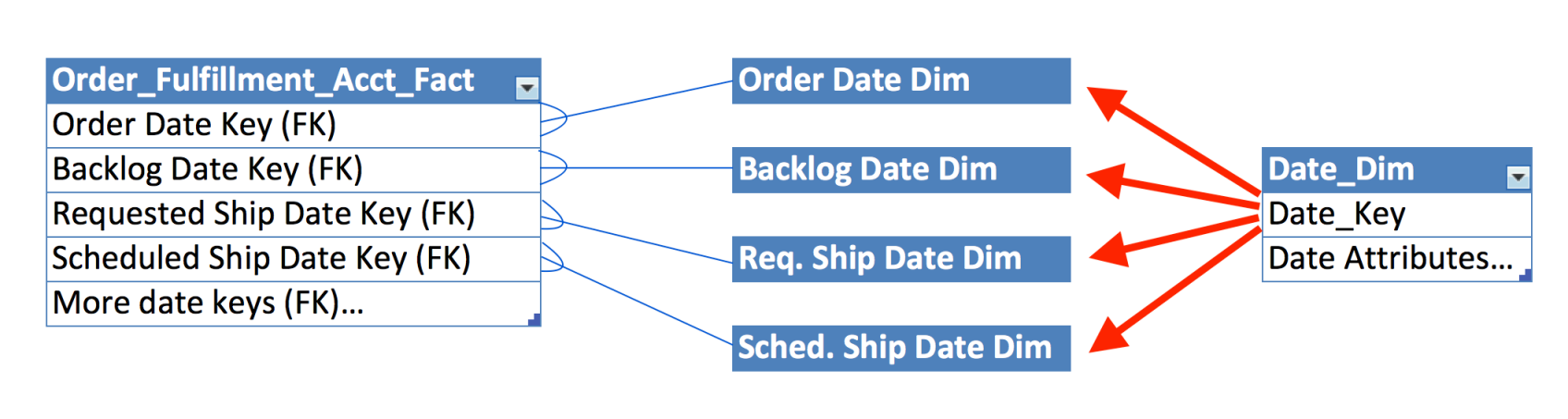
Surrogate Key: Implementation Considerations

- Most databases have a way of generating a surrogate key for a table
- Most ETL tools can generate a surrogate key with a counter as well
- An ETL generated key works no matter what type of database under the hood
- Database surrogate key is easier to implement
- Special care is required not to reset the database surrogate key sequence

Date Dimension Considerations

- Predictable stable dimension
- Key assigned chronically
- YYYYMMDD instead of surrogate sequence number
  - Query YYYYMMDD to bypass Date dimension table is not advised
  - Useful for partitioning large fact table
- Need default (dummy) row/key for unknown or to be determined date
- Handle time-of-day separate from Date dimension for day parts
- Consider transaction date/time stamp as fact

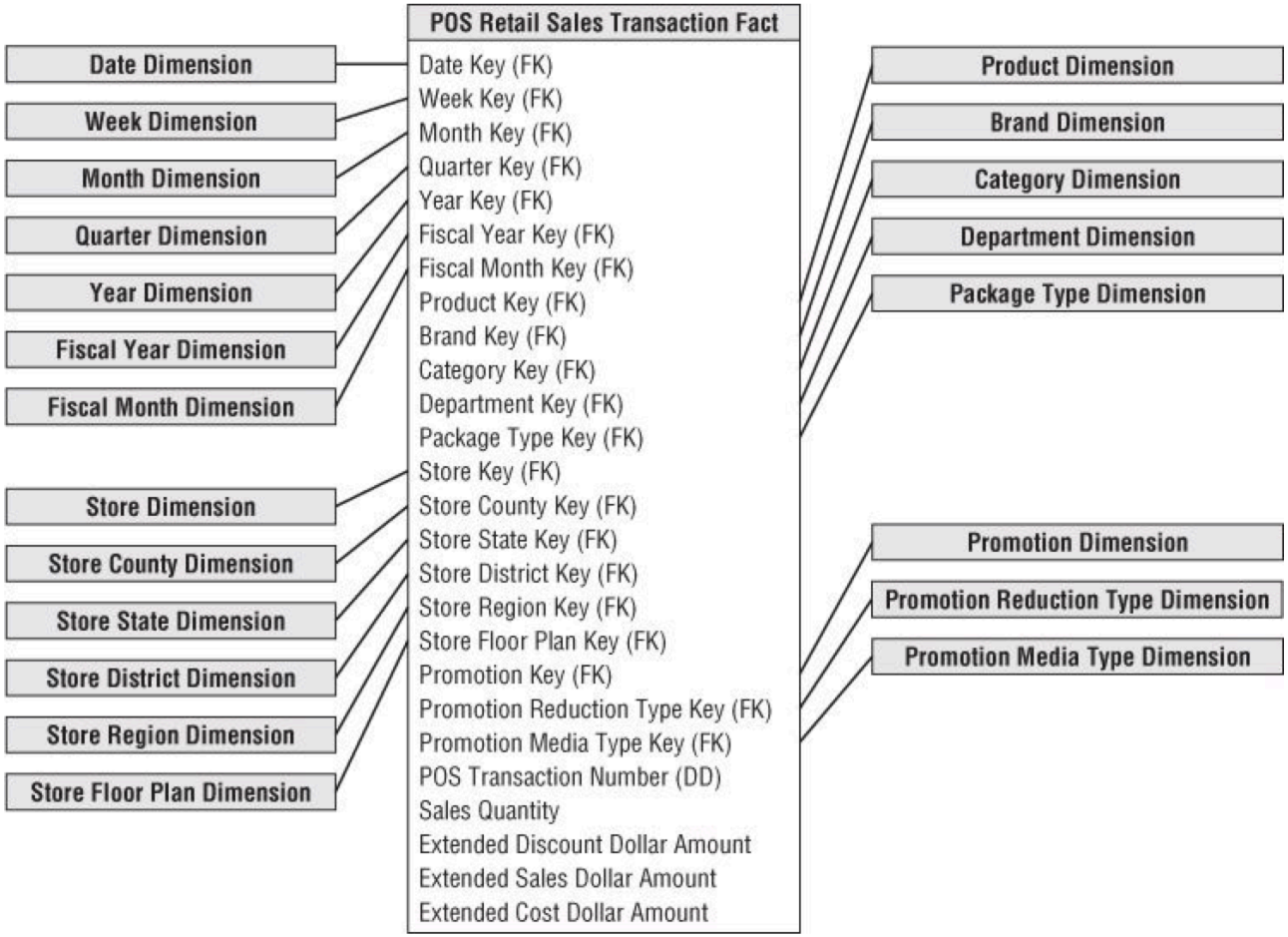
Roll-Playing Dimensions



- A single physical dimension table playing multiple logical roles
- Options for implementing roll-playing dimensions
  - Database views
  - Aliases or synonyms of a single physical table
  - Aliasing within the BI tool's semantic layer
- Other examples: Employee (Cashier/Associate), Location (Origin/Destination), Physicians (Primary/Referring), etc.

Avoid Too Many Dimension "Centipede" Fact Tables

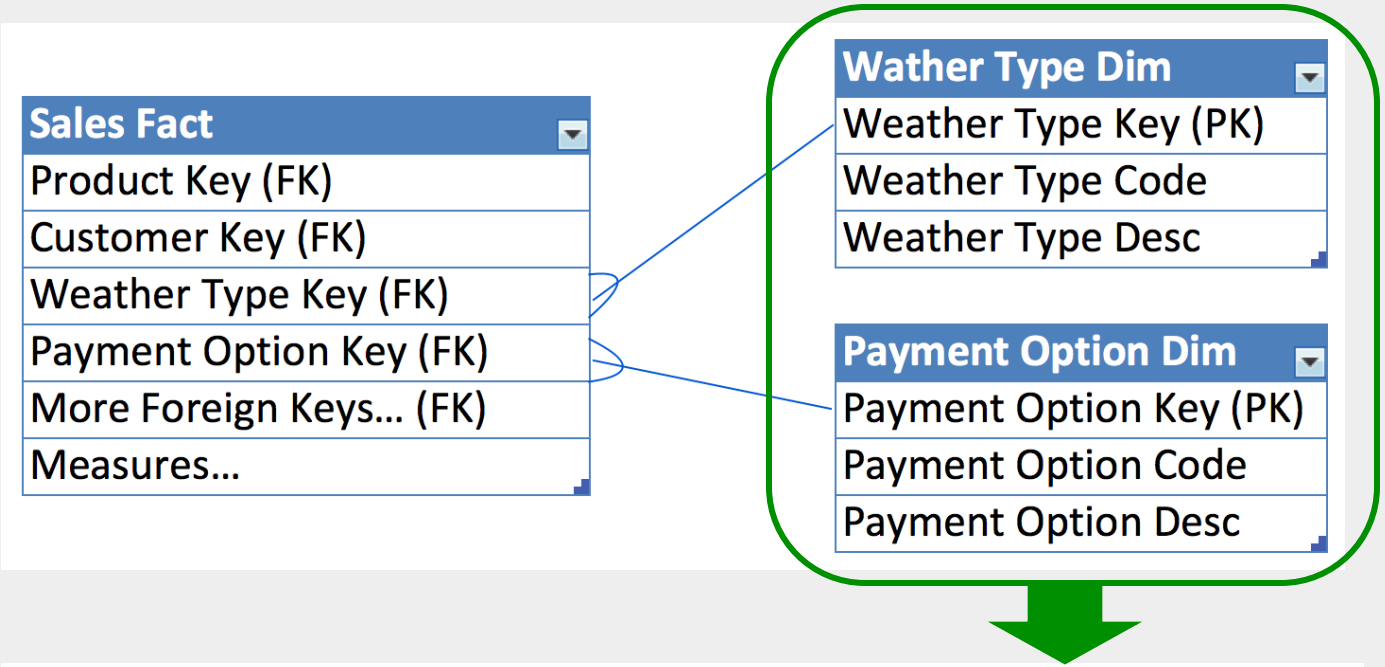
Figure 3.17 Centipede fact table with too many normalized dimensions.

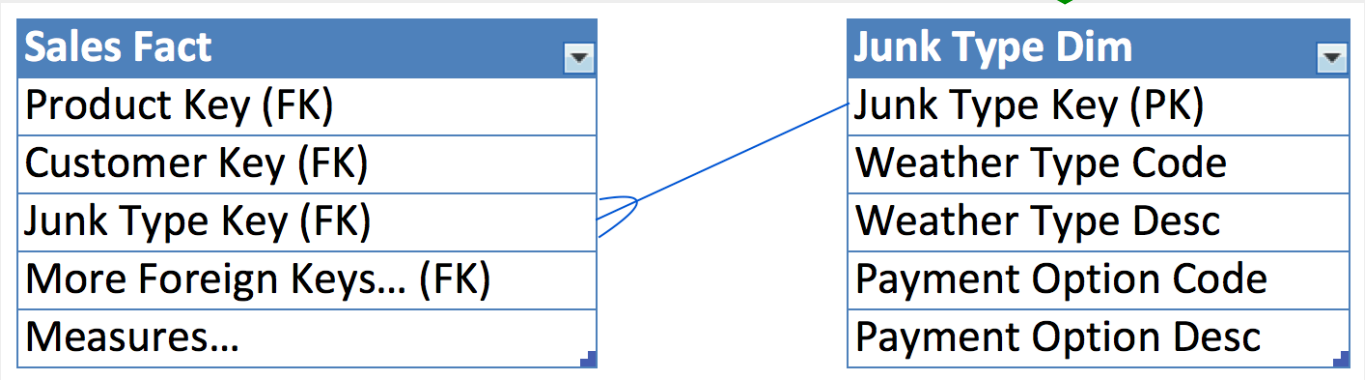


Junk Dimensions

- Combine miscellaneous transaction flags/ indicators into junk dimension
- Potentially less desirable alternative:
  - Multiple fact table keys to low-cardinality dimensions
- Undesirable alternatives:
  - Place flags/indicators directly in fact table as text facts or DDs
  - Place flags/indicators in transaction dimension

Junk Dimensions (cont'd)





Dealing with Nulls: Operations with Nulls

- Operations with Null can be tricky sometimes

```
SELECT 1 + null col
  FROM dual; -- null

SELECT CASE WHEN null = null THEN 1 ELSE 2 END AS col
  FROM dual; -- 2

SELECT CASE WHEN null IS null THEN 1 ELSE 2 END AS col
  FROM dual; -- 1

SELECT CASE WHEN 1 IS null THEN 1 ELSE 2 END AS col
  FROM dual; -- 2

SELECT DECODE (null, null, 1, 2, 3) as col
  FROM dual; -- 1

SELECT DECODE (1, null, 1, 2, 3) as col
  FROM dual; -- null
```

Dealing with Nulls: Operations with Nulls (Cont'd)

- Operations with Null can result in missing information

```
SELECT deptid, SUM(annual_salary) AS annual_salary_by_dept
  FROM (
        SELECT deptid, empid, month_pay*12+bonus AS annual_salary
          FROM (
                SELECT 'd01' AS deptid, 'e0001' AS empid,
                       10000 AS month_pay, 1000 AS bonus
                  FROM dual UNION ALL
                SELECT 'd01' AS deptid, 'e0002' AS empid,
                      100000 AS month_pay, null AS bonus
                  FROM dual
               )
       )
 GROUP BY deptid
;

DEPTID ANNUAL_SALARY_BY_DEPT
------ --------------------
d01                    121000 -- $1,200,000 missing!
```

Dealing with Nulls: Operations with Nulls (Cont'd)

- Aggregate functions handle Null gracefully

```
SELECT room_no, AVG(math) AS avg_math_by_room, AVG(writing) AS
avg_writing_by_dept
  FROM (
        SELECT room_no, student_id, math, writing
          FROM (
                SELECT 'rm2001' AS room_no, 's0001' AS student_id,
                        80 AS math,   90 writing
                  FROM dual UNION ALL
                SELECT 'rm2001' AS room_no, 's0002' AS student_id,
                       100 AS math, null writing
                  FROM dual
               )
       )
 GROUP BY room_no;

ROOM_NO AVG_MATH_BY_ROOM AVG_WRITING_BY_DEPT
------- ---------------- -------------------
rm2001                90                  90
```

Dealing with Nulls: Rule of Thumb in Data Warehousing

- NULL fact table foreign keys
  - No NULL is allowed as it breaks referential integrity
  - Substitute key to special dimension row (a.k.a. dummy dimension row)
- NULL dimension attributes
  - Strongly discouraged to avoid unexpected query results (e.g. invalidating index strategy)
  - Use default values instead – N/A, Unknown, Invalid, To be determined,…
- NULL facts
  - Use ONLY IF it truly means N/A, Unknown, and Invalid, not zero

Slowly Changing Dimensions Revisited

- Changes in dimensions arrive
  - Unexpectedly, sporadically, and far less frequently than fact table measurements → Slow changing

- Type 1: Overwrite

Supplier_Key	Supplier_Code	Supplier_Name	Supplier_State
123	ABC	Acme Supply Co	CA

Supplier_Key	Supplier_Code	Supplier_Name	Supplier_State
123	ABC	Acme Supply Co	IL

Slowly Changing Dimensions Revisited

- Type 2: Insert a new dimension row with the new data and new effective date

Supplier_Key	Supplier_Code	Supplier_Name	Supplier_State	Start_Date	End_Date
123	ABC	Acme Supply Co	CA	2000-01-01	2004-12-21

Type 2

Supplier_Key	Supplier_Code	Supplier_Name	Supplier_State	Start_Date	End_Date
123	ABC	Acme Supply Co	CA	2000-01-01	2004-12-21
124	ABC	Acme Supply Co	IL	2004-12-22	2999-12-31

- Type 1/Type 2 Hybrid
  - Most common hybrid

Customer_Key	Customer_ID	Customer_Birth	Supplier_State	Start_Date	End_Date
332	C01	1977-08-01	CA	2000-01-01	2004-12-21

Type 1 Type 2

Customer_Key	Customer_ID	Customer_Birth	Supplier_State	Start_Date	End_Date
332	C01	1977-09-01	CA	2000-01-01	2004-12-21
333	C01	1977-09-01	IL	2004-12-22	2999-12-31

Week 04 Class Exercise

- Junk Dimension Implementation Demo
- SCD Type 1 / Type 2 / Hybrid Demo

Week 04 Topic: Dimensional Modeling: Basic Dimension Tables Techniques

Questions?