Data Warehouse Design Techniques – Snapshot Fact Tables

‹   ›

♡ 2

Last week we defined the three types of fact tables and have established a solid foundation in the definition and use of transactional fact tables. This week we will focus on periodic snapshot fact tables.

Snapshot Fact Tables

(Periodic) Snapshot fact tables capture the state of the measures based on the occurrence of a status event or at a specified point-in-time or over specified time intervals (week, month, quarter, year, etc.).

```
           FACT_MONTHLY_SALES
P  *  SALE_MONTH_DATE_KEY       NUMBER
P  *  SALE_CUSTOMER_KEY         NUMBER
P  *  SALE_PRODUCT_KEY          NUMBER
P  *  SALE_AUTHOR_GROUP_KEY     NUMBER
P  *  SALE_STORE_KEY            NUMBER
      SALE_QTY                  NUMBER
      SALE_AMT                  NUMBER▼
```

The Snapshot Fact table and the **Transactio**nal fact table can be very similar in design. In the design above the sales are rolled up to the month in the Periodic Snapshot fact table. In this table, you are unable to drill down past the month, as the month is the lowest grain of this Periodic Snapshot fact table. Periodic Snapshot tables are also important when measuring those metrics which may not have a corresponding **transactio**n, such as air temperature. We cannot capture **transactio**ns causing the air temperature to rise or fall but we can measure the temperature at predetermined points in time.

```
          FACT_TEMPERATURE
P  *  FT_LOACATION_KEY
P  *  FT_CLOUD_COVER_KEY
P  *  FT_DATE
      FT_TEMPERATURE
      FT_HUMIDITY
```

As you can see **transactio**n fact tables can fall short in the needs of the users to measure key metrics and the use of snapshot facts are one of the ways to address these **transactio**nal fact shortcomings.

Snapshot Fact Design

Snapshot fact tables are similar to the **transactio**nal fact table in design but sample the data at predetermined points in time or as a result of a specific event. The snapshot fact in dense as there will be a record for each time period or event regardless of the number of **transactio**ns or amount of change in the measure.

The traditional example used for a snapshot fact can be seen using bank account balances. A **transactio**n fact will capture each deposit or withdrawal when it occurs but the **transactio**n fact does not care about the balance of the account.

Account	Transaction Date	Transaction Type	Transaction Amount
12345	3/1/2017	Initial Deposit	$1,000
12345	3/3/2017	Payroll Deposit	$500
12345	3/3/2017	ATM Withdrawal	$100
12345	3/7/2017	Check	$75
12345	3/10/2017	Payroll Deposit	$500
12345	3/11/2017	ATM	$250
12345	3/17/2017	Payroll Deposit	$500
12345	3/20/2017	Check	$110
12345	3/22/2017	Web Payment	$135
12345	3/24/2017	Payroll Deposit	$500
12345	3/24/2017	Web Payment	$90
12345	3/24/2017	ATM Withdrawal	$125
12345	3/28/2017	Check	$50
12345	3/30/2017	ATM Withdrawal	$65
12345	3/31/2017	Payroll Deposit	$500

The snapshot fact, on the other hand, calculates the balance of the account at the end of each day. This snapshot fact can then be easily used to calculate the average daily balance for interest or fees.

Account	Date	Balance
12345	3/1/2017	$1,000
12345	3/2/2017	$1,000
12345	3/3/2017	$1,400
12345	3/4/2017	$1,400
12345	3/5/2017	$1,400
12345	3/6/2017	$1,400
12345	3/7/2017	$1,325

12345	3/8/2017	$1,325
12345	3/9/2017	$1,325
12345	3/10/2017	$1,825
12345	3/11/2017	$1,575
12345	3/12/2017	$1,575
12345	3/13/2017	$1,575
12345	3/14/2017	$1,575
12345	3/15/2017	$1,575
12345	3/16/2017	$1,575
12345	3/17/2017	$2,075
12345	3/18/2017	$2,075
12345	3/19/2017	$2,075
12345	3/20/2017	$1,965
12345	3/21/2017	$1,965
12345	3/22/2017	$1,830
12345	3/23/2017	$1,830
12345	3/24/2017	$2,115
12345	3/25/2017	$2,115
12345	3/26/2017	$2,115
12345	3/27/2017	$2,115
12345	3/28/2017	$2,065
12345	3/29/2017	$2,065
12345	3/30/2017	$2,000
12345	3/31/2017	$2,500

The measurements in snapshot fact tables are not additive, able to add them together into some meaningful metric, but these measures are usually semi-additive, able to use aggregate functions like AVG, MIN and MAX. For example, the average daily balance ($1,734.03) is quickly computed by adding all of the balance columns and dividing by the number of days or simply using the AVG aggregate function.

Capturing Active Record for Event or Time Period

Have you ever run up against a requirement which states that 1) you must be able to allow a record to be updated for an event or time period in the past and 2) you must be able to reproduce the detailed report as it was on any given day. This second requirement prevents one from simply updating the measures within a fact table, which is frowned upon in dimensional modeling as it impacts the reports upon which decisions were made. Therefore, we need to find a way to capture this information and associate it with the correct event or time period. In the example below, we have an equipment status report fact. Here we see that initially a record was entered on February 1 with a quantity of 5.

ORGANIZATION KEY	EQUIPMENT KEY	MONTH KEY	DATE KEY	QTY	ACTIVE RECORD FLAG	ACTIVE MONTHLY RECORD FLAG
111	2345	201702	20170201	5	1	1

Notice how we have added an additional column called active monthly record flag. This will indicate if this is the record which should be used on all current and future calculations for this organization, equipment and month.

Later in the month on February 24 the equipment is re-inventoried and 2 more pieces of equipment are now available. Here we need to inactivate the original record as both the active record and the active monthly record and make the newly inserted fact record on February 24 become the active record and the active monthly record for reporting. Here we have captured when the record changed so if we need to show what a report printed on February 10 looked like we could regenerate this report.

ORGANIZATION KEY	EQUIPMENT KEY	MONTH KEY	DATE KEY	QTY	ACTIVE RECORD FLAG	ACTIVE MONTHLY RECORD FLAG
111	2345	201702	20170201	5	0	0
...	...	...	...	...	...	...
111	2345	201702	20170224	7	1	1

On the first of each month, the current active record should be copied to the next month as the active record and the previous active record should be made inactive. Please Note: the active monthly record flag does not change this record remains active as this will be used when computing monthly summaries.

ORGANIZATION KEY	EQUIPMENT KEY	MONTH KEY	DATE KEY	QTY	ACTIVE RECORD FLAG	ACTIVE MONTHLY RECORD FLAG
111	2345	201702	20170201	5	0	0
. . .	. . .	. . .	. . .	. . .	. . .	. . .
111	2345	201702	20170224	7	0	1
. . .	. . .	. . .	. . .	. . .	. . .	. . .
111	2345	201703	20170301	7	1	1

Now on March 4 they realized that there was one more piece of equipment on hand and should have been accounted for in February but sat on someone's desk for several days without being input. To address this challenge, we need to go back and update February's count without impacting the report as it was delivered on February 28 or March 1. To address this requirement, we need to do some work:

1) add a record on March 4 for the updated February record-setting the active monthly record to true and the active record flag to false

2) inactivate the active monthly record inserted on February 24

3) insert a new March active March record

4) inactivate the previously active March record.

Now the snapshot fact contains all detailed transactions and we can easily calculate the month over month values from this fact table.

ORGANIZATION KEY	EQUIPMENT KEY	MONTH KEY	DATE KEY	QTY	ACTIVE RECORD FLAG	ACTIVE MONTHLY RECORD FLAG
111	2345	201702	20170201	5	0	0
. . .	. . .	. . .	. . .	. . .	. . .	. . .
111	2345	201702	20170224	7	0	0
. . .	. . .	. . .	. . .	. . .	. . .	. . .
111	2345	201703	20170301	7	0	0
. . .	. . .	. . .	. . .	. . .	. . .	. . .
111	2345	201702	20170304	8	0	1
111	2345	201703	20170304	8	1	1

You can see the power of this functionality for detailed **transactio**nal reporting as well as the ease in which data can be summarized for time period reporting.

Next week we will study the usage and benefits of accumulating fact table.
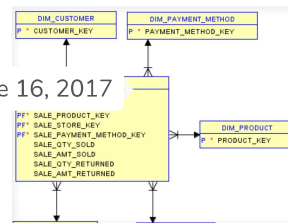
Jim McHugh

Related posts

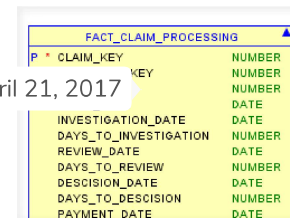| July 28, 2017	July 5, 2017	June 16, 2017	May 24, 2017	April 21, 2017
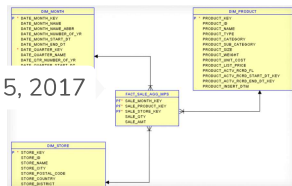
Loading Hierarchical

Data Warehouse Design Techniques – Constraints and Indexes

:: Read more

Data Warehouse Design Techniques – Aggregates

:: Read more

Data Warehouse Design Techniques – Derived Schemas

:: Read more

Bridge Tables

:: Read more

Data Warehouse Design Techniques – Accumulating Snapshot Fact Tables

:: Read more

---

1 Comment

hadi ismanto says:						Reply

May 9, 2018 at 3:32 am

Thank you for the lovely explanation

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website

☐

Save my name, email, and website in this browser for the next time I comment.

Post Comment

About us

NuWave delivers Analytics and Data Management solutions to customers in public sector markets.

Our Vision

NuWave's vision is to be in the Top Ten small businesses in the Washington, DC region in terms of customer satisfaction, quality of services and growth. We will be considered a leading edge company providing analytics and data management solutions to public sector customers.

Our Mission

Our mission is to provide innovative, customized solutions for customer needs through development, selection and integration of leading edge technologies. We give our customers the ability to communicate and manage information by using the strength of our visionary insight, technical expertise, and commitment to teamwork.

Get in Touch

NuWave Solutions, LLC
8000 Westpark Drive, Suite 640
McLean, VA 22102
P: (703) 790-1122
E: marketing@nuwavesolutions.com