3.4 Slowly Changing Dimensions

Temporal data warehousing is often known as *Slowly Changing Dimensions* or *SCD*. Slowly changing dimensions are dimensions where the records of these dimensions change slowly over a period of time. Using the bookshop case study earlier in this chapter, the Book dimension has price information, and it is common that the price of a book changes "slowly" over time; and thats why the Book dimension is an example of a slowly changing dimension. It is called slowly because then changes of price are not, for example, done every hour, or even every week. Over a longer period of time, such as months or even years, the price of a book is changed.

The topic of slowly changing dimensions is different from attributes or records that are "rapidly" changed, such as the location attribute of a moving taxi, which records changes very frequently (e.g. the change of location coordinate is recorded every 1 minute in the operational database), or the price of shares in the stock market, which may change every a few second, etc. The topic of these rapid changes of records or values is often studied in the area of Real-Time Data Warehousing or Stream Data Warehousing, which is not the scope of this chapter.

Originally, there are three different types of treatment of SCD, called Type 1, Type 2, and Type 3. Each of these types treats an implementation of SCD differently. However, lately, data warehousing practitioners have added with new types, called Type 0, Type 4, and Type 6, which enrich the implementation options for SCD.

3.4.1 SCD Type 0 and Type 1

SCD Type 0 and Type 1 are quite similar; they do not actually record the history of changes in the dimension.

In **Type 0**, the dimension stores the "Original or Initial" value of the records, when the data warehousing is built. If the value of the dimension attributes changes, the changes are not recorded. The records remain the same as when the records were first inserted into the data warehouse.

Using the Bookshop case study, these values are the "Full Price". For many books, the original or initial price was the full price; but for some books, the price listed initially may not be the full price. So, BookDim table using SCD Type 0, which lists the full prices, is shown in Table 3.13. If the book prices change after that, the new price will not be recorded in the data warehouse.

Table 3.13 BookDim (SCD Type 0)

BookID	Book Title	Author	Price
C1	CSIRO Diet	CSIRO Team	$45.95
H6	Harry Potter 6	Rowling	$30.95
DV	Da Vinci Code	Dan Brown	$27.95
...	...	...	...

For other systems, it is more desirable to store the original or initial value, rather than the so called full price as in this example.

Because SCD Type 0 does not record the history of book price, the star schema does not have any temporal dimension, hence, the star schema for the Bookshop case study is the one shown in Figure 3.1.

In **Type 1** does not record the history of changes either. It only records the latest value of the record. Using the BookDim example, the book price in the BookDim will be the latest price. This means that when there is a change of price, the old price will be overwritten by the new price in the BookDim table. Table 3.14 shows the contents of the BookDim table using SCD Type 1. In this case, the $10.00 price of Book ID H6 is the latest price. Note that the other two books in this example have the same price as the full price in SCD Type 0, but that does not mean that the prices were never changed. This BookDim table only tells us that these are the current price of the books.

Table 3.14 BookDim (SCD Type 1)

BookID	Book Title	Author	Price
C1	CSIRO Diet	CSIRO Team	$45.95
H6	Harry Potter 6	Rowling	**$10.00**
DV	Da Vinci Code	Dan Brown	$27.95
...	...	...	...

Like SCD Type 0, SCD Type 1 star schema does not maintain a temporal dimension, and hence the star schema is the one shown in Figure 3.1.

Because SCD Type 0 and Type 1 do not maintain the history of book prices, if we need to produce a report that joins BookDim with BookSalesFact tables, we need to be careful not to draw an association between the book price from BookDim and the TimeID from the BookSalesFact, because the book price in the report may not necessarily the book price at that particular TimeID. In such a report, the column title for the book price can be changed to ”original book price” (for SCD Type 0), or ”latest book price” (for SCD Type 1), in order to avoid any misunderstanding in interpreting the report.

3.4.2 SCD Type 2

SCD Type 2 keeps track the history, but not separating the history from the main dimension; instead the new records keep added to the dimension. Using the BookDim as an example, when the price of a book is changed, it creates ”another book” with the same details, except with the new Book ID, and of course with the new price. In addition to the StartDate and EndDate, it also has a CurrentFlag, to indicate whether a record is the current record or a past record. Any additional information, such as the remarks, may also be included. The BookDim table for SCD Type 2 is shown in Table 3.15.

Table 3.15 BookDim (SCD Type 2)

BookID	Book Title	Author	StartDate	EndDate	Price	Remarks	CurrentFlag
C1_1	CSIRO Diet	CSIRO Team	Jan2007	July2007	$45.95	Full Price	N
C1_2	CSIRO Diet	CSIRO Team	Aug2007	Oct2007	$36.75	20% Discount	N
C1_3	CSIRO Diet	CSIRO Team	Nov2007	Jan2008	$23.00	Half Price	N
C1_4	CSIRO Diet	CSIRO Team	Feb2008	Dec9999	$45.95	Full Price	Y
H6_1	Harry Potter 6	Rowling	Jan2007	Mar2007	$21.95	Launching	N
H6_2	Harry Potter 6	Rowling	Apr2007	Jan2008	$30.95	Full Price	N
H6_3	Harry Potter 6	Rowling	Feb2008	Dec9999	$10.00	End of Product Sale	Y
DV_1	Da Vinci Code	Dan Brown	Jan2007	Dec9999	$27.95	Full Price	Y
...	...	...	...	...	...	...	...

Note that the same book has a different BookID for different book price and period. Normally, the BookID attribute is implemented as a Surrogate Key; but in this example, we just added with a sequence number to the original BookID to differentiate the same book in different time period. Because of these multiple BookID for the same book, the report that joins the BookSalesFact and the BookDim will look like as follows (see Report 3 SCD Type 2 in Table 3.16).

Note that for example, the first record in Report 3, the BookID is C1_4, because this is the BookID of CSIRO Diet book in March 2008. Because SCD Type 2 only changes the original BookDim dimension, the star schema looks similar to that of Figure 3.1, but BookDim has additional attributes. See Figure 3.8.

3.4.3 SCD Type 3

SCD Type 3 is a simplification of Type 2. Unlike Type 2 that maintains multiple records of the same book, Type 3 does not have multiple records for the same book. One book has one entry in the BookDim table. For the price, it only records the current price and the previous price. In other words, it does not maintain the entire history of price changes; it only keeps the last two prices. The BookDim table for SCD Type 3 is shown in Table 3.17. Note that for the book that does not have any previous price, a Null value will be recorded in the Previous Price column.

The main rationale for adopting SCD Type 3 is that most of the analysis will be based on the current price and at most one past price, which is the previous price. Perhaps, this is for comparison with the trend when the price was the previous price. It is assumed that analysing the complete history is not necessary.

Additionally, although we could store when the price was changed, the original SCD Type 3 does not record this. It only records the current and the previous prices. Without keeping the date when the price was changed, it is not possible then to correlate the book price with the TimeID information in the BookSalesFact table. Consequently, the report that can be produced will need to particularly pay attention

Table 3.16 Report 3 (SCD Type 2)

TimeID	BranchID	BookID	BookTitle	Author	Price	Number_Of_Books_Sold
Mar2008	City	**C1_4**	CSIRO Diet	CSIRO Team	$45.95	5
Mar2008	City	**H6_3**	Harry Potter 6	Rowling	$10.00	15
Mar2008	City	**DV_1**	Da Vinci Code	Dan Brown	$27.95	23
Mar2008	City	...	...	...	...	...
Mar2008	Chadstone	**C1_4**	CSIRO Diet	CSIRO Team	$45.95	15
Mar2008	Chadstone	**H6_3**	Harry Potter 6	Rowling	$10.00	3
Mar2008	Chadstone	**DV_1**	Da Vinci Code	Dan Brown	$27.95	2
Mar2008	Chadstone	...	...	...	...	...
Mar2008	Camberwell	**C1_4**	CSIRO Diet	CSIRO Team	$45.95	1
Mar2008	Camberwell	**H6_3**	Harry Potter 6	Rowling	$10.00	1
Mar2008	Camberwell	**DV_1**	Da Vinci Code	Dan Brown	$27.95	2
Mar2008	Camberwell	...	...	...	...	...
Mar2008	...	...	...	...	...	...
...	...	...	...	...	...	...
...	...	...	...	...	...	...
Dec2007	City	**C1_3**	CSIRO Diet	CSIRO Team	$23.00	15
Dec2007	City	**H6_2**	Harry Potter 6	Rowling	$30.95	6
Dec2007	City	**DV_1**	Da Vinci Code	Dan Brown	$27.95	6
Dec2007	City	..	...	...	...	...
Dec2007	Chadstone	**C1_3**	CSIRO Diet	CSIRO Team	$23.00	10
Dec2007	Chadstone	**H6_2**	Harry Potter 6	Rowling	$30.95	8
Dec2007	Chadstone	**DV_1**	Da Vinci Code	Dan Brown	$27.95	1
Dec2007	Chadstone	..	...	...	...	...
Dec2007	Camberwell	**C1_3**	CSIRO Diet	CSIRO Team	$23.00	18
Dec2007	Camberwell	**H6_2**	Harry Potter 6	Rowling	$30.95	3
Dec2007	Camberwell	**DV_1**	Da Vinci Code	Dan Brown	$27.95	2
Dec2007	Camberwell	..	...	...	...	...
Dec2007	...	...	...	...	...	...
...	...	...	...	...	...	...

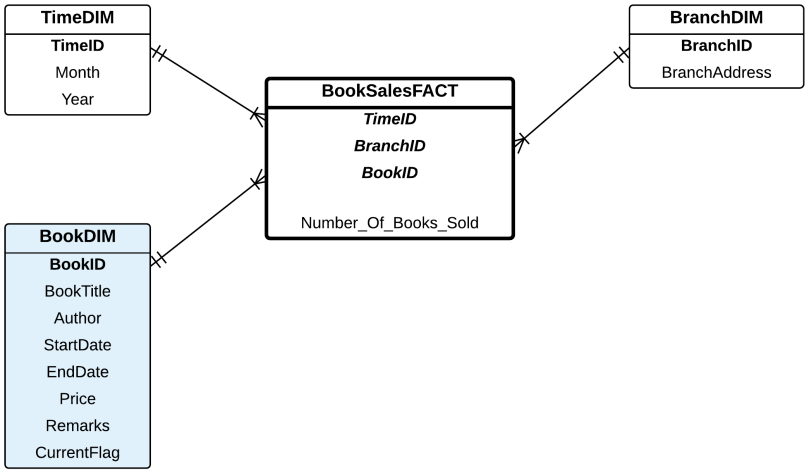

Figure 3.8 The Bookshop Star Schema with SCD Type 2 for BookDim

Table 3.17 BookDim (SCD Type 3)

BookID	Book Title	Author	CurrentPrice	PreviousPrice
C1	CSIRO Diet	CSIRO Team	**$45.95**	**$23.00**
H6	Harry Potter 6	Rowling	**$10.00**	**$30.95**
DV	Da Vinci Code	Dan Brown	**$27.95**	**Null**
...	...	...	...	...

to the potential mismatch between the information in the book price column (from the BookDim table) and the TimeID column (from the BookSalesFact table)

The star schema for SCD Type 3 is shown in Figure 3.9.

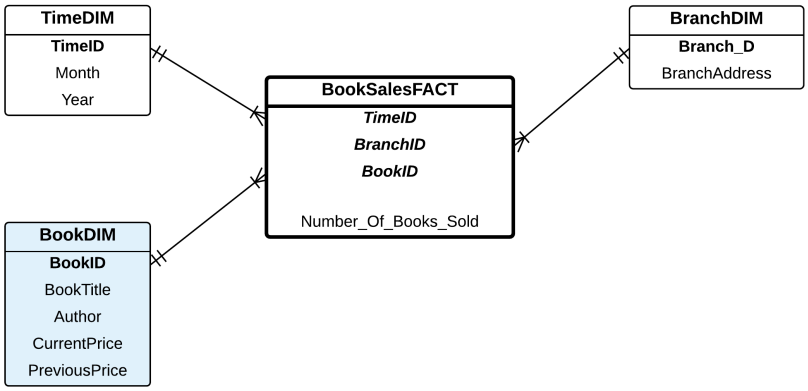

Figure 3.9 The Bookshop Star Schema with SCD Type 3 for BookDim

3.4.4 SCD Type 4

SCD Type 4 is a new way to treat SCD that was not in the original SCD theory. In SCD Type 4, we create a new dimension to maintain the history of attribute value change. Basically, the temporal dimension that is described in the beginning of this chapter is SCD Type 4. In Type 4, the original BookDim is kept without the price attribute. The price attribute (and StartDate and EndDate) are separated into another table; this is the BookPriceDim table. The BookDim and the BookPriceDim tables are in Tables 3.18 and 3.19.

Table 3.18 BookDim Table

BookID	BookTitle	Author
C1	CSIRO Diet	CSIRO Team
H6	Harry Potter 6	Rowling
DV	Da Vinci Code	Dan Brown
...	...	...

The main advantage of Type 4 is that we do not need to have a different BookID for the same book. Additionally, the entire history of changes is kept. As shown earlier, this method will guarantee that the report that joins the information from the

Table 3.19 BookPriceDim Table

BookID	StartDate	EndDate	Price	Remarks
C1	Jan2007	July2007	$45.95	Full Price
C1	Aug2007	Oct2007	$36.75	20% Discount
C1	Nov2007	Jan2008	$23.00	Half Price
C1	Feb2008	Dec9999	$45.95	Full Price
H6	Jan2007	Mar2007	$21.95	Launching
H6	Apr2007	Jan2008	$30.95	Full Price
H6	Feb2008	Dec9999	$10.00	End of Product Sale
DV	Jan2007	Dec9999	$27.95	Full Price
...	...	...	...	...

BookSalesFact table and the dimension tables will be accurate reflecting the correct book price at certain TimeID. The star schema for SCD Type 4 is shown previous in Figure 3.2.

3.4.5 SCD Type 6

SCD Type 6 is actually a combination between Type 2 and Type 3. In Type 3, only the current price and the previous price are recorded; not the entire history. In Type 2, the entire history of changes is maintained, but a separate identifier (e.g. Surrogate Key is needed). In Type 6, a separate identifier for the same book is not needed (like Type 3), but the entire history is kept (like Type 2). SCD Type 6 for the BookDim table is shown in Table 3.20.

Table 3.20 BookDim (SCD Type6)

BookID	Book Title	Author	StartDate	EndDate	CurrentPrice	PreviousPrice	Remarks	Cu
C1	CSIRO Diet	CSIRO Team	Jan2007	July2007	$45.95	Null	Full Price	
C1	CSIRO Diet	CSIRO Team	Aug2007	Oct2007	$36.75	$45.95	20% Discount	
C1	CSIRO Diet	CSIRO Team	Nov2007	Jan2008	$23.00	$36.75	Half Price	
C1	CSIRO Diet	CSIRO Team	Feb2008	Dec9999	$45.95	$23.00	Full Price	
H6	Harry Potter 6	Rowling	Jan2007	Mar2007	$21.95	Null	Launching	
H6	Harry Potter 6	Rowling	Apr2007	Jan2008	$30.95	$21.95	Full Price	
H6	Harry Potter 6	Rowling	Feb2008	Dec9999	$10.00	$30.95	End of Product Sale	
DV	Da Vinci Code	Dan Brown	Jan2007	Dec9999	$27.95	Null	Full Price	
...	...	...	...	...	...	...	...	

In Type 6, there is no need to maintain a separate history table. The history itself is kept in the original dimension table. The star schema for SCD Type 6 is shown in Figure 3.10. Note that the BookDim table has a composite key comprising of BookID, StartDate, and EndDate.

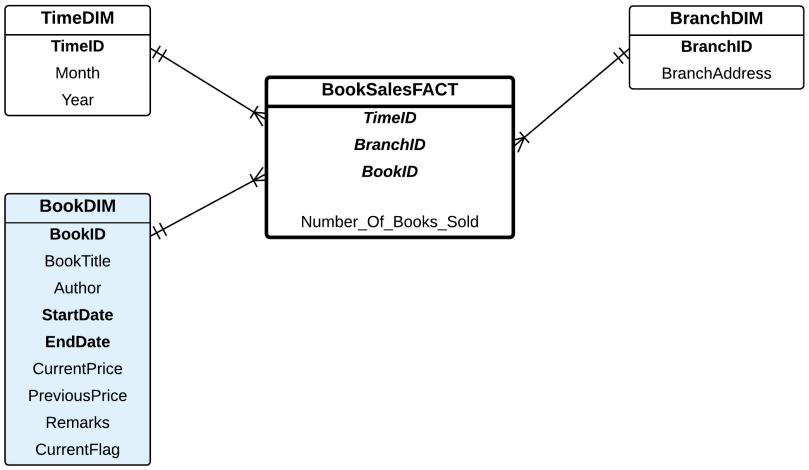

Figure 3.10 The Bookshop Star Schema with SCD Type 6 for BookDim

3.4.6 Implementation of SCD in SQL

The concepts of SCD are very easy to digest. However implementing SCD in SQL has come challenging technical aspects. In this section, we are going to examine the SQL to create the six SCD types.

The source tables in the operational database are the Book Table, and the BookPrice-History Table, as shown in the E/R diagram (refer to Figure 3.1). The structures of these two tables are:

- **Book** (**BookID**, BookTitle, Author)
- **BookPriceHistory** (**BookID, StartDate, EndDate**, Price, Remarks)

SCD Type 0 is a non-temporal dimension. Book dimension table using SCD Type 0 is shown in Table 3.13. Note that the ”original price” may not the be ”initial price”, because some books have a launching price which can be heavily discounted. Therefore, the Book dimension table using SCD Type 0 checks for the Remarks ”Full Price”.

```
create table SCD0 as
```

```
select distinct
   B.BookID, B.BookTitle,
   B.Author, H.Price as OriginalPrice
from Book B, BookPriceHistory H
where B.BookID = H.BookID
and H.Remarks = 'Full Price';
```

SCD Type 1 uses the latest price (refer to Table 3.14). We could choose the book where the End Date is 'Dec9999'. Alternatively, we can sort the Start Date, and choose the "latest" Start Date. In the following SQL, the inner query ranks the book records based on the Start Date in a descending order, and the outer query choose those books which have rank 1.

```
create table SCD1 as
select
   T.BookID, T.BookTitle,
   T.Author, T.Price as CurrentPrice
from (
  select
     B.BookID, B.BookTitle, B.Author,
     to_date(H.StartDate, 'MonYYYY'), H.Price,
     rank() over( partition by B.BookID
        order by to_date(H.StartDate, 'MonYYYY') desc)
        as Rank
  from Book B, BookPriceHistory H
  where B.BookID = H.BookID) T
where T.Rank = 1;
```

SCD Type 2 is basically a join between Book Table and Book Price History Table, so that we can get the book details, as well as the Start Date and End Date as well. However, the book with the same Book ID should be added with a sequence number, as previously shown in Table 3.15.

The SQL for SCD Type 2 uses the rank function as well as the partition clause, so that the same book will be ranked according to their Start Date. The Current Flag column will identify whether the book record has the current price or not.

```
create table SCD2 as
select  B.BookID  || '_' ||
   rank() over(partition by B.Book_ID
     order by to_date(H.StartDate, 'MonYYYY') asc)
     as BookID,
  B.BookTitle, B.Author, H.StartDate,
  H.EndDate, H.Price, H.Remarks,
  case H.End_Date When 'Dec9999' then 'Y' else 'N'
  end as CurrentFlag
from Book B, BookPriceHistory H
```

```
where B.BookID = H.BookID;
```

SCD Type 3 is quite complex, as it has not only Current Price (like SCD Type 2) but also Previous Price. If SCD Type 2 uses rank 1, SCD Type 3 needs rank 2 (for the Previous Price) as well. Hence, SCD Type 3 joins the table with rank 1 and the table with rank 2. Since some books do not have Previous Price, we need to use an outer join instead of an inner join.

```
create table SCD3 as
select
   T1.BookID, T1.BookTitle, T1.Author,
   T1.CurrentPrice, T2.CurrentPrice as PreviousPrice
from (
   select
      T.BookID, T.BookTitle,
      T.Author, T.Price as CurrentPrice
   from (
      select
         B.BookID, B.BookTitle,
         B.Author, to_date(H.StartDate, 'MonYYYY'),
         H.Price,
         rank() over( partition by B.BookID
            order by to_date(H.StartDate, 'MonYYYY') desc)
            as Rank
      from Book B, BookPriceHistory H
      where B.BookID = H.BookID) T
   where T.Rank = 1) T1,
   (select
      T.BookID, T.BookTitle,
      T.Author, T.Price as CurrentPrice
   from (
      select B.BookID, B.BookTitle, B.Author,
         to_date(H.StartDate, 'MonYYYY'), H.Price,
         rank() over( partition by B.BookID
            order by to_date(H.StartDate, 'MonYYYY') desc)
            as Rank
      from Book B, BookPriceHistory H
      where B.BookID = H.BookID) T
   where T.Rank = 2) T2
where T1.BookID = T2.BookID(+);
```

SCD Type 4 is actually the Temporal Data Warehousing discussed in the first part of this chapter. The SQL to create the Book Dimension is an entire extraction from the Book Price History Table from the operational database.

```
create table SCD4 as
```

```
select * from BookPriceHistory;
```

SCD Type 6 is a combination between SCD Type 2 and SCD Type 3. Hence, we can simply join these two SCD Types. Remember that SCD Type 2 identifier (Book ID) is different from the original Book ID used by SCD Type 3. Therefore, we cannot simply use an equi-join between SCD Type 2 and SCD Type 3. Instead, we need to check if the Book ID of SCD Type 3 is part of the Book ID of the SCD Type 2, using the "Like" operator in SQL.

```
create table SCD6 as
select
   SCD3.BookID, SCD3.BookTitle, SCD3.Author,
   SCD2.StartDate, SCD2.EndDate,
   SCD3.CurrentPrice, SCD3.PreviousPrice,
   SCD2.Remarks, SCD2.CurrentFlag
from SCD2, SCD3
where SCD2.BookID LIKE SCD3.BookID||'_%';
```

3.5 Summary

In this chapter, we focus on incorporating historical data in the data warehouse. This is called *Temporal Data Warehousing*. A temporal data warehousing uses the concept of the Bridge Table (or a Weak Entity), where the history is maintained in a bridge table. Maintaining the history of certain attributes is important in order to make associative analysis more accurate when analysing the reports produced by the fact and dimensions. However, certain degree of caution when joining the fact table and the temporal dimension, especially when the level of granularity of time between the fact and the temporal dimension is not the same.

Temporal data warehousing is also known as *Slowly Changing Dimensions* (*SCD*). In this chapter, various treatment and types for SCD are presented. Different types will server different purposes of the data warehousing.

EXERCISES

3.1 The "Auto Car Service" performs car services for their customers. Every time a service is conducted, a record is entered into the database. The information recorded includes service number, service name, date, car registration number, the staff who handled the service and liaised with the customer, the mechanic who performed the repair, and total cost of the repair.

You are required to build a data warehouse to analyze the number of services for each car model, mechanic, month/year, and part. Note that every service may use several different parts. The price of each part may from time to time changes, and