Data Warehousing Slow Changing Dimensions (SCD)

White Paper

CLOUDBASIC

Slow Changing Dimensions Implementation in CloudBasic

This white paper deals with how CloudBasic handles Slow Changing Dimensions (SCD), that is, changes occurring over time to the context data of the data mart; it assumes familiarity with the subject of dimensional modeling in data warehouses, but also briefly describes the different SCD types based on the nature of the changes and business needs for history record keeping.

We can see a dimension data row as having two parts, the first part is data attributes that are static and do not change over time, such as the data that identifies a row uniquely. The second part is the data attributes that change over time, but not too frequently, and that may or not need history tracking.

On the history tracking topic, the specific business need may be to not track the history of those changes at all, fully track those changes or just partially track them by keeping current and previous data values for the dimension; in data warehouse parlance, these history tracking strategies are referred to as SCD Types 1, 2 and 3, respectively.

SCD Type 1: Overwrite

This dimension type updates the contents by replacing the old values; at the row level, new data inserts a new row; and the update operation may have a change detection method to only update columns with changed values or simply replaces the entire data contents, in any case, the data value history is lost.

SCD Type 2: Full Changes History

In this type new content again simply creates a new row; and in order to keep track of content changes, data updates also add a new row, and a versioning technique is used to indicate what the current row is.

A general recommendation for tracking these changes is that at minimum three additional attributes be created: a timestamp that indicates the creation of the row, a second timestamp that indicates the update operation, and finally a current row indicator; however, in practice a data row version number may be sufficient, where the highest version value indicates the current row.

SCD Type 3: Current and Previous Values

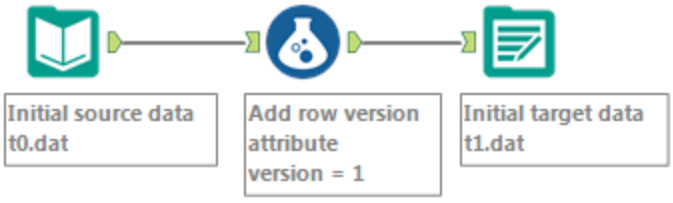
As before, with this type new content simply creates a new row; but in this case the concern is to keep only the current and previous content; therefore, there is only need for creating an additional attribute value.

Seeding

There is an initial step whereby a starting target data mart replica is created from the source data mart; CloudBasic is aware of what the unique identifier for each data row is, which remains static over time (commonly known as primary key), and treats the rest of the data in the row as the slow changing dimension.

SCD Type 2

In order to implement a SCD Type 2, CloudBasic creates a version number for each row; upon insert this value is 1, and subsequent updates increase this value by 1.



SCD Type 2 Seeding Process

Source Table

pk	f1	f2	f3
1	a	b	c
2	d	e	f
3	g	h	i

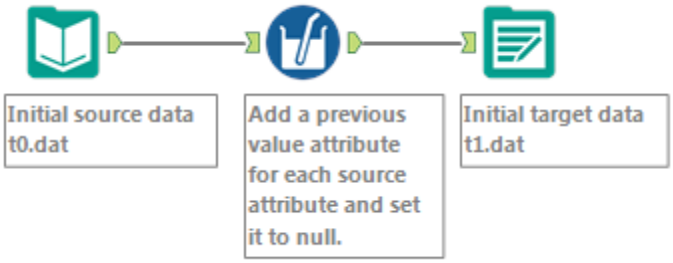
→

Target Table

pk	v	f1	f2	f3
1	1	a	b	c
2	1	d	e	f
3	1	g	h	i

SCD Type 3

On the other hand, to implement a SCD Type 3 and keep the partial history of these data attributes, CloudBasic creates additional attributes to store the immediately last known data value.



SCD Type 3 Seeding Process

Source Table

pk	f1	f2	f3
1	a	b	c
2	d	e	f
3	g	h	i

→

Target Table

pk	f1	f2	f3	_f1	_f2	_f3
1	a	b	c	null	null	null
2	d	e	f	null	null	null
3	g	h	i	null	null	null

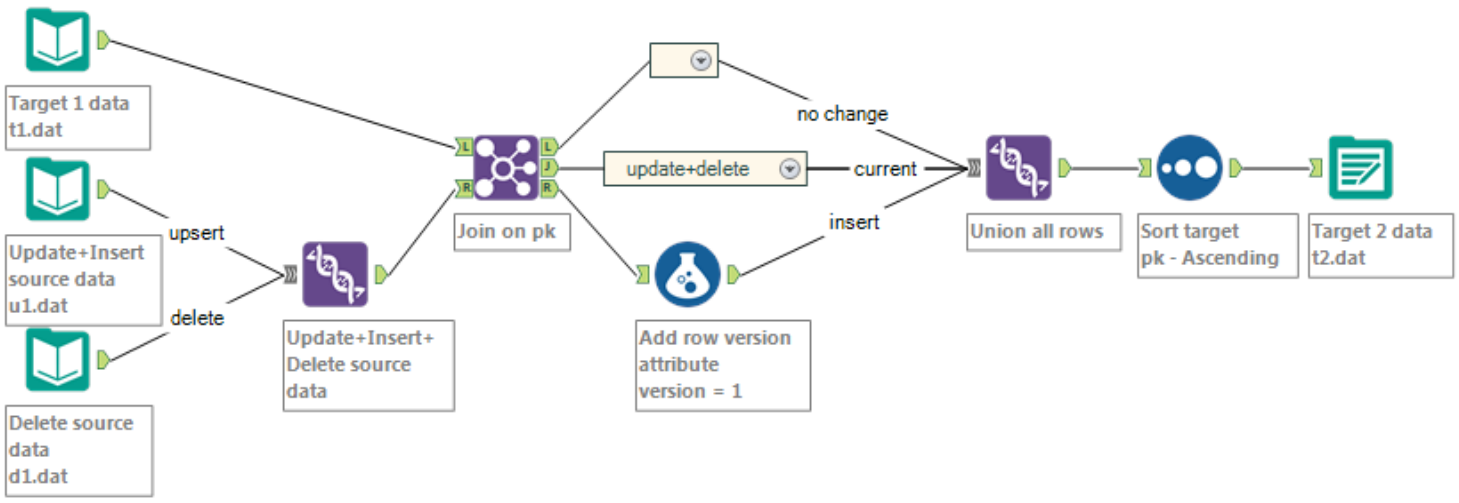
Applying Data Changes

In SCD Type 2 the content change processing keeps track of all of the updates to the data attributes; that is, the full history available, and a version number is used for this purpose.

1. For updates, CloudBasic locates the corresponding rows, reads their version number, increments this number by 1 and inserts a new row with the changed data.

2. For deletes, the corresponding rows are located, their version number is read, incremented by 1 and a row with empty data values is inserted.

3. For inserts, the version number attribute is added to the new rows, and then inserted.



SCD Type 2 Content Change Process

Changed Data

pk	f1	f2	f3
1	j	b	c
2	null	null	null
4	k	l	m

→

SCD Type 2 Target Table

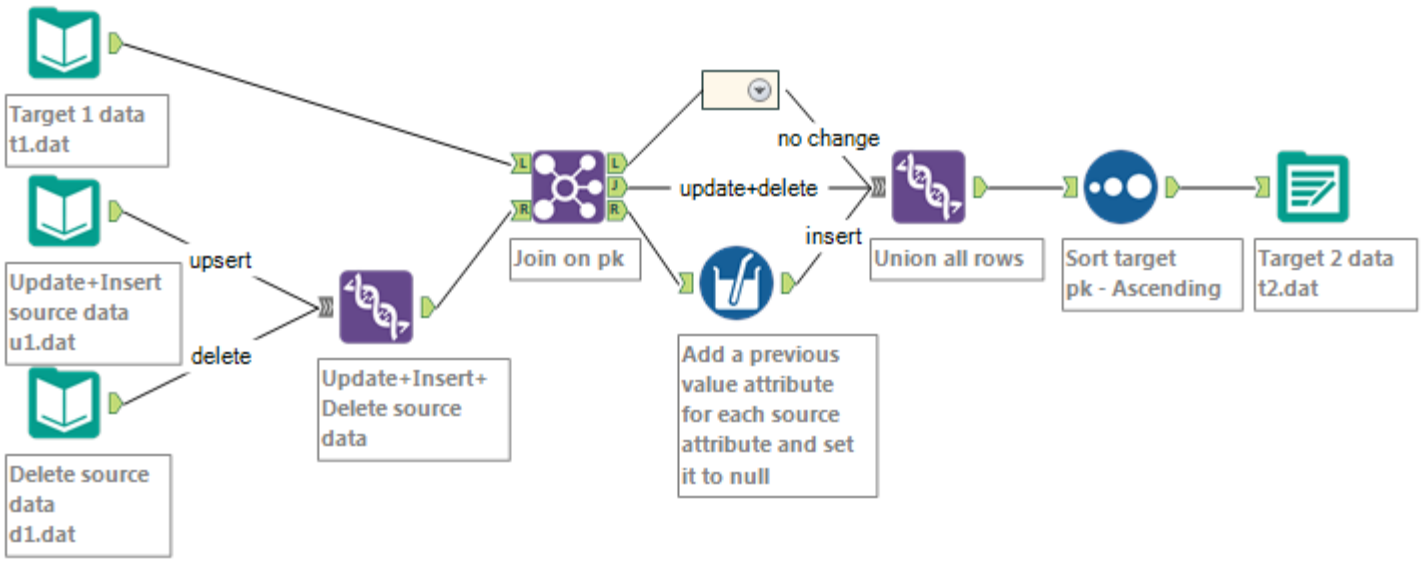
pk	v	f1	f2	f3
1	1	a	b	c
1	2	j	b	c
2	1	d	e	f
2	2	null	null	null
3	1	g	h	i
4	1	k	l	m

In SCD Type 3 the content change processing keeps track of the current and the immediately previous value of the data attributes; that is, there is only one level of history available.

4. For updates, CloudBasic locates the corresponding rows and moves their current data values into the previous data values, then copies the updated data into the current data values.

5. For deletes, the corresponding rows are located and their current data values are moved into the previous data values, then the current data values are emptied.

6. For inserts, additional attributes are created to store the immediately last known data value, and then inserted.



SCD Type 3 Content Change Process

Changed Data

pk	f1	f2	f3
1	j	b	c
2	null	null	null
4	k	l	m

→

SCD Type 3 Target Table

pk	f1	f2	f3	_f1	_f2	_f3
1	j	b	c	a	b	c
2	null	null	null	d	e	f
3	g	h	i	null	null	null
4	k	l	m	null	null	null

Managing Data Structure Changes

Business needs also change over time, and an application data structure may change due to this; CloudBasic is aware of changes made to data structures and handles them accordingly.

SCD Type 2

Changed Data

pk	f1	f2	f3	f4
3	null	null	null	null
4	k	l	m	n
5	o	p	q	r

↓

SCD Type 2 Target Table

Pk	v	f1	f2	f3	f4
1	1	a	b	c	null
1	2	j	b	c	null
2	1	d	e	f	null
2	2	null	null	null	null
3	1	g	h	i	null
3	2	null	null	null	null
4	1	k	l	m	null
4	2	k	l	m	n
5	1	o	p	q	r

SCD Type 3

Changed Data

pk	f1	f2	f3	f4
3	null	null	null	null
4	k	l	m	n
5	o	P	q	r

↓

SCD Type 3 Target Table

pk	f1	f2	f3	f4	_f1	_f2	_f3	_f4
1	J	b	c	null	a	b	c	null
2	Null	null	null	null	d	e	f	null
3	Null	null	null	null	g	h	i	null
4	K	l	m	n	k	l	m	null
5	O	p	q	r	null	null	null	null