Возьмите в работу скрипт инкрементальной загрузки SCD1 и доработайте его до загрузки SCD2.

Что меняется:

- 1. В приемнике появляются три поля SCD2:
 - effective from dttm
 - effective to dttm
 - deleted flg
- 2. Алгоритм вставки не меняется;
- 3. При обновлении добавляется 1 строка и дополнительно необходимо закрыть старую версию;
- 4. При удалении добавляется 1 строка с deleted_flg = 1 (или 'Y') и effective_to_dttm = технической бесконечности и закрывается старая версия.

Также можно ориентироваться на пример в excel-файле

Дополнено:

Хорошо!

```
Вот здесь не хватает фильтра
-- 6. Обновим данные effective_to_dt по измененным значениям (val) или запись удалена
Тест-кейс такой - удалить строку у которой больше одной версии
-- 0.2.4
/*
delete from snow_source where 1=1;
insert into snow_source values (1,'A',to_date('01-09-2021','DD-MM-YYYY'));
insert into snow_source values (2,'B',to_date('01-09-2021','DD-MM-YYYY'));
insert into snow_source values (4,'D',to_date('04-09-2021','DD-MM-YYYY'));
commit;
*/
На стейджинге не создаем новые столбцы (deleted_flg), в промежуточный объект лучше.
```

•	Output 	DE2TM.SNOW_	STAGE_DEL ×	■ DE2TM.SNO	W_STAGE 3 >	× III DE2TM	.SNOW_STAGE_DEL 2	× ≡ D	E2TM.SNOW_TARGET ×	
I<	< 7 rows ~	> > G	= + - -			✓ 5 DDL	*			
	I≣ ID ≑	III VAL ≑	■ EFFECTIV	E_FROM_DTTM		■ EFFECTIV	E_TO_DTTM		■ DELETED_FLG ÷	
1			2021-09-01	00:00:00.000	900	2022-02-15	23:59:59.000000			
2			2022-02-16	00:00:00.000	900	5999-12-31	00:00:00.000000			
3			2021-09-01	00:00:00.000	900	5999-12-31	00:00:00.000000			
4			2021-09-01	00:00:00.000	900	2021-09-04	23:59:59.000000			
5			2021-09-05	00:00:00.000	900	2022-02-15	23:59:59.000000			
6			2022-02-16	00:00:00.000	900	5999-12-31	00:00:00.000000			
7			2021-09-04	00:00:00.000	900	5999-12-31	00:00:00.000000			

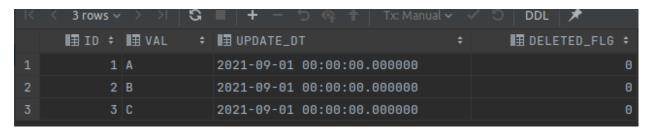
Этапы:

1:

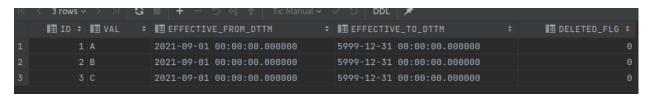
id	value	update dt	jd	yalue	effective from dt	effective to dt	deleted flg
	1 A	01.09.2021		1 <u>A</u>	01.09.2021	31-12-5999	0
	2 <u>B</u>	01.09.2021		2 B	01.09.2021	31-12-5999	0
	3 Ç	01.09.2021		3 Ç	01.09.2021	31-12-5999	0

insert into snow_source values (1,'A',to_date('01-09-2021','DD-MM-YYYY')); insert into snow_source values (2,'B',to_date('01-09-2021','DD-MM-YYYY')); insert into snow_source values (3,'C',to_date('01-09-2021','DD-MM-YYYY')); commit;

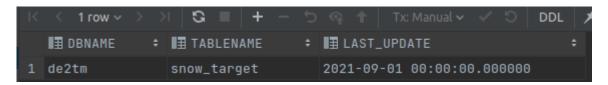
stage:



target:



meta:

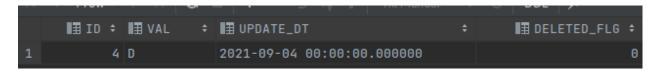


2:

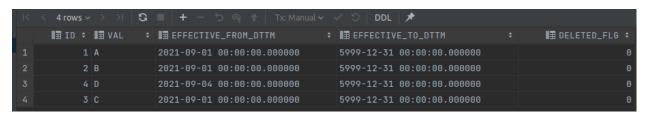
<u>lid</u>	value	update_dt	jd	yalue	effective_from_dt	effective to dt	deleted_flg
	1 <u>A</u>	01.09.2021		1 💆	01.09.2021	31-12-5999	0
	2 <u>B</u>	01.09.2021		2 <u>B</u>	01.09.2021	31-12-5999	0
	3 <u>C</u>	01.09.2021		3 <u>C</u>	01.09.2021	31-12-5999	0
	4 <u>Q</u>	04.09.2021		4 D	04.09.2021	31-12-5999	0

insert into snow_source values (4,'D',to_date('04-09-2021','DD-MM-YYYY')); commit;

stage:



target:



meta:

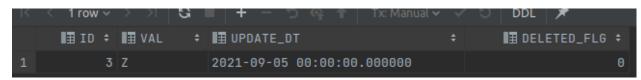
	■ DBNAME	■ TABLENAME	■ LAST_UPDATE :
1	de2tm	snow_target	2021-09-04 00:00:00.000000

3:

id	yalue	update dt	jd	value	effective from dt	effective to dt	deleted flg
	1 A	01.09.2021		1 A	01.09.2021	31-12-5999	0
	2 B	01.09.2021		2 B	01.09.2021	31-12-5999	0
	3 Z	05.09.2021		3 Ç	01.09.2021	04.09.2021	0
	4 D	04.09.2021		4 D	04.09.2021	31-12-5999	0
				3 Ž	05.09.2021	31-12-5999	0

delete from snow_source where 1=1; insert into snow_source values (1,'A',to_date('01-09-2021','DD-MM-YYYY')); insert into snow_source values (2,'B',to_date('01-09-2021','DD-MM-YYYY')); insert into snow_source values (3,'Z',to_date('05-09-2021','DD-MM-YYYY')); insert into snow_source values (4,'D',to_date('04-09-2021','DD-MM-YYYY')); commit;

stage:



target:

Тест-кейс такой - удалить строку у которой больше одной версии

ID	VAL	EFFECTIVE_FROM_DTTM	EFFECTIVE_TO_DTTM	DELETED_FLG
3	Z	05-SEP-21 12.00.00.000000 AM	31-DEC-99 12.00.00.000000 AM	0
1	Α	01-SEP-21 12.00.00.000000 AM	31-DEC-99 12.00.00.000000 AM	Θ
2	В	01-SEP-21 12.00.00.000000 AM	31-DEC-99 12.00.00.000000 AM	Θ
3	C	01-SEP-21 12.00.00.000000 AM	04-SEP-21 11.59.59.000000 PM	1
4	D	04-SEP-21 12.00.00.000000 AM	31-DEC-99 12.00.00.000000 AM	Θ

meta:

5 rows selected.

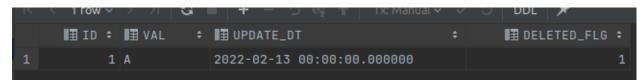
	II DBNAME :	II TABLENAME :	■ LAST_UPDATE	‡
1	de2tm	snow_target	2021-09-05 00:00:00.000000	

4:

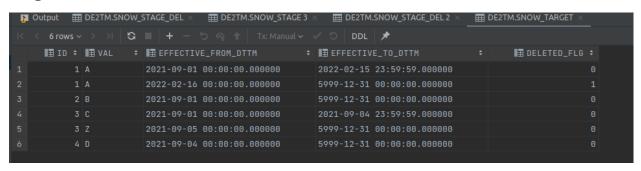
jġ	value	update_dt	jd	yalue	effective_from_dt	effective to dt	deleted_flg
	2 B	01.09.2021		1 A	01.09.2021	current date-1	0
	3 <u>Z</u>	05.09.2021		2 B	01.09.2021	31-12-5999	0
	4 <u>D</u>	04.09.2021		3 Ç	01.09.2021	04.09.2021	0
				4 D	04.09.2021	31-12-5999	0
				3 <u>Z</u>	05.09.2021	31-12-5999	0
				1 Å	current_date	31-12-5999	1

```
delete from snow_source where 1=1; insert into snow_source values (2,'B',to_date('01-09-2021','DD-MM-YYYY')); insert into snow_source values (3,'Z',to_date('05-09-2021','DD-MM-YYYY')); insert into snow_source values (4,'D',to_date('04-09-2021','DD-MM-YYYY')); commit;
```

stage (пишу удаленные, чтоб был единый алгоритм обработки):



target:



5.

Тест-кейс такой - удалить строку у которой больше одной версии + На стейджинге не создаем новые столбцы (deleted_flg), в промежуточный объект лучше.

-- 0.2.5 Тест-кейс такой - удалить строку у которой больше одной версии. Удалим 3 строку (Z) /*
delete from snow_source where 1=1;
insert into snow_source values (2,'B',to_date('01-09-2021','DD-MM-YYYY'));
insert into snow_source values (4,'D',to_date('04-09-2021','DD-MM-YYYY'));
commit;
*/

