

T2

T1

updates1 ~~insert~~ insert, delete, updates2 ~~insert~~

Order

Status

--	--	--

New
PAid
Cancelled
Delivered

status_history

o-id	status	author	time

insert into order (.....) values (..... New...) → previous
 update → gone ...
 update →

set status = PAid
 set status = Delivered

trigger

s1 ~~insert~~ insert, delete, update
s2 ~~insert~~

before
after

insert
update
delete

6

before

current.
new.

after

old.
current.

```
CREATE TRIGGER log_changes
AFTER UPDATE ON employees
FOR EACH ROW
BEGIN
    INSERT INTO employees_log (employee_id, name, action)
    VALUES (OLD.employee_id, OLD.name, 'updated');
END;
```



```
-- january
create table shares
(
  action text,
  amount integer,
  name text
);
```

```
-- february
alter table shares add column price integer;
```

```
-- ...
alter table shares add column info text;
```

+ JAVA.

v1

table version

client1

curr v2
v3...v15

curr v15
[]

client N
curr v7
v8..v15

curr v15
[]

v2

v3

v15

+ JAVA

- 1) each structure change in ONE file (different)
- 2) curr version
- 3) all versions, filter ($v \rightarrow v > \text{current}$). for each ($v \rightarrow \text{apply}(v)$)

- + library
- + curr state
- + sql 1... sql n (versions)
- + filter + apply missing.

'0' 1
 10
 '9' 2
 3

V 001
 :
 V 999

V 2024 11 27 -- comment .sql

2024 11 28

32
 33