**ERROR TRACKING DASHBOARD**

**FILTERS:**

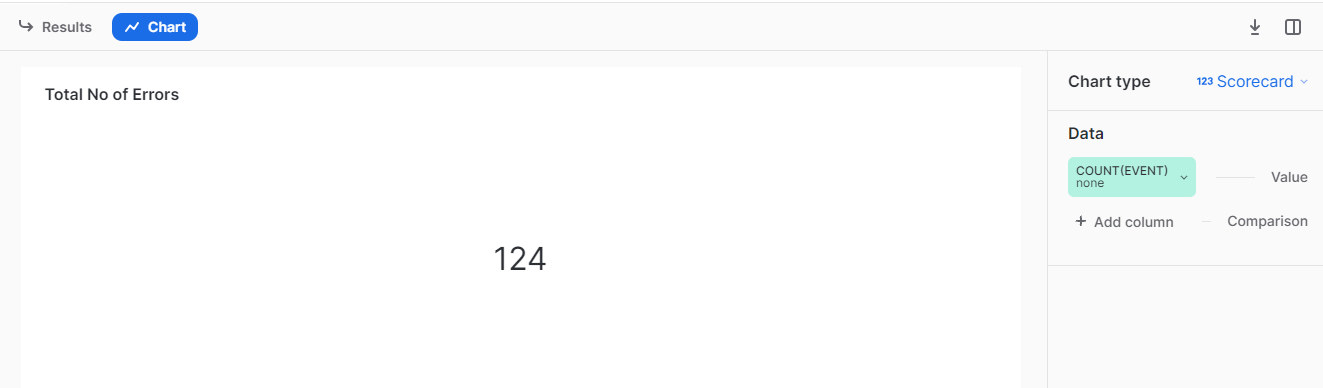
* **DATERANGE(Default Filter)**

**2.1 Total No of error**

SELECT COUNT(EVENT)

FROM FIVETRAN.FIVETRAN\_LOG.LOG

WHERE EVENT='WARNING' OR EVENT='SEVERE' AND \_fivetran\_synced=:daterange;

****

**2.2 Error Rate %**

SELECT

round( (SUM(CASE

WHEN event = 'SEVERE' Or event='WARNING'

THEN 1

ELSE 0

END)/

SUM(CASE

WHEN message\_event = 'sync\_end'

THEN 1

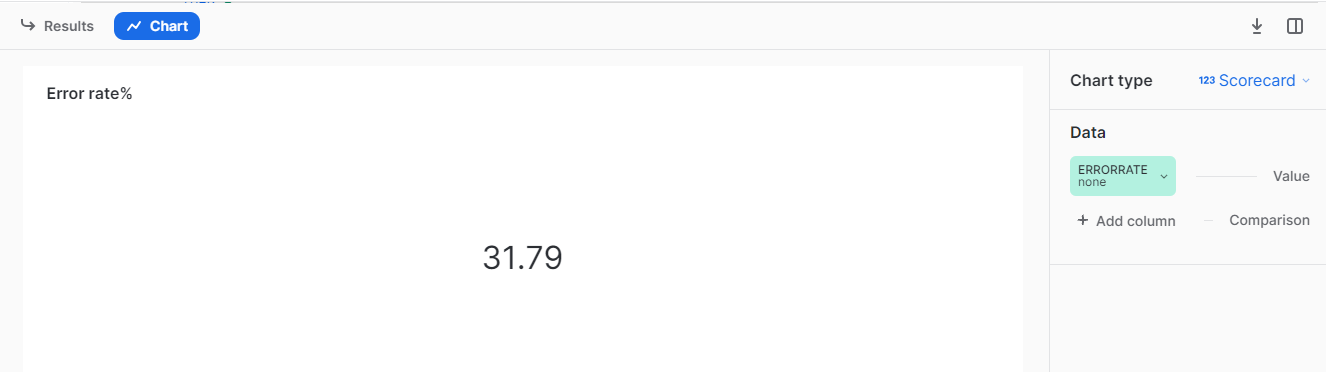
ELSE 0

END) )\*100,2 )AS ERRORRATE

FROM

FIVETRAN.FIVETRAN\_LOG.LOG

WHERE \_fivetran\_synced=:daterange;



**2.3 Trend Of unsuccessful sync**

select date\_trunc('day',time\_stamp) as date,count(event) as sync\_count,split\_part(parse\_json(message\_data):status,':',1) as status

from FIVETRAN.FIVETRAN\_LOG.LOG

where event='SEVERE' or event='WARNING' AND \_fivetran\_synced=:daterange

group by date\_trunc('day',time\_stamp),status;

select (SYNC\_ID) as sync\_count,parse\_json(message\_data):status as Status,\_fivetran\_synced

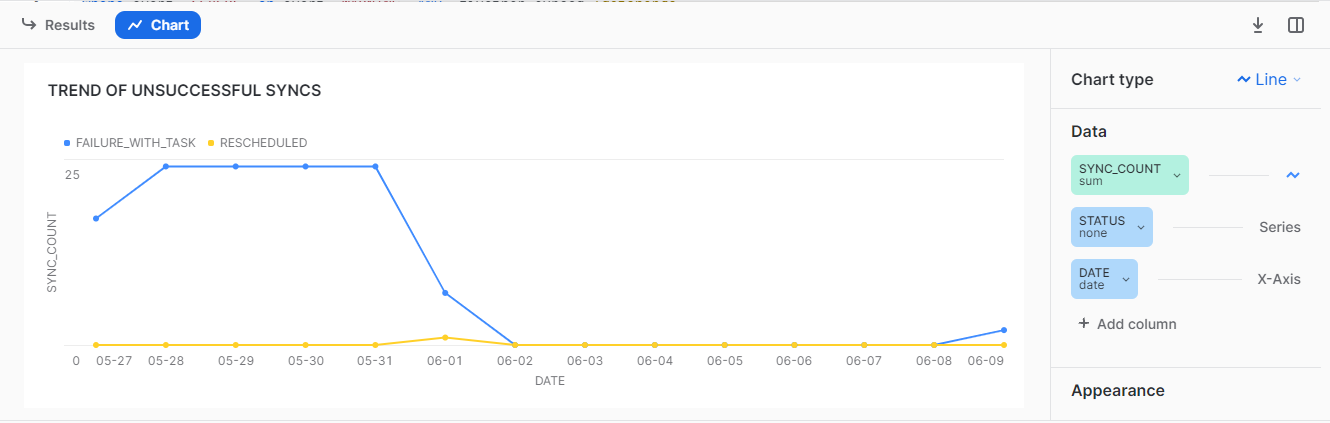
from fivetran.fivetran\_log.log

where message\_event='sync\_end' AND EVENT = 'SEVERE' or event='WARNING' ;

group by sync\_count,

group by Status,

group by\_fivetran\_synced;



**2.4 Common error messages**

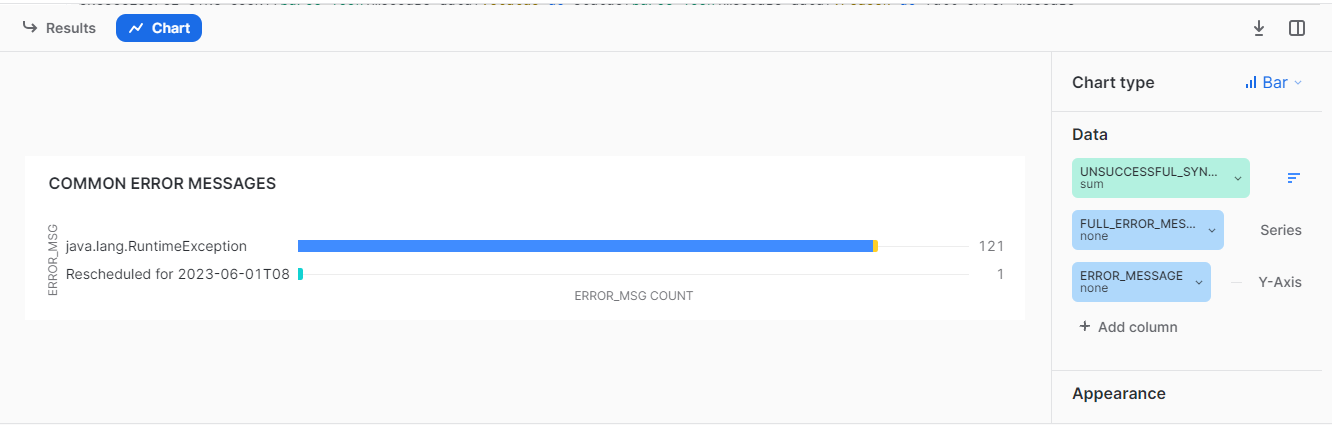
select distinct split\_part(parse\_json(message\_data):reason,':',1) As Error\_message, COUNT(parse\_json(message\_data):reason) as UNSUCCESSFUL\_SYNC\_COUNT,parse\_json(message\_data):status as Status,parse\_json(message\_data):reason as full\_error\_message

from "FIVETRAN"."FIVETRAN\_LOG"."LOG"

where event = 'SEVERE'or event='WARNING' AND \_fivetran\_synced=:daterange

group by connector\_id,parse\_json(message\_data):status,

parse\_json(message\_data):reason;



**2.5 Error rate per Connector**

SELECT

top 7 connector\_id,

COUNT(CASE WHEN event IN ('SEVERE', 'WARNING') THEN 1 END) AS event\_count,

COUNT(CASE WHEN message\_event = 'sync\_end' THEN 1 END) AS total\_sync,

case when total\_sync >0 then ROUND( (event\_count/total\_sync)\*100,2) end as ERROR\_RATE

FROM

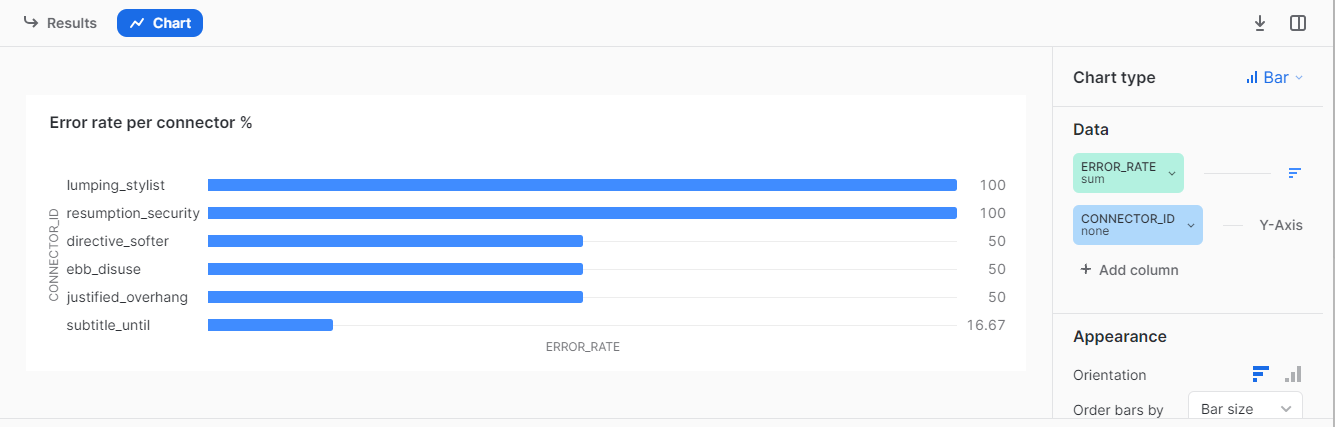
FIVETRAN.FIVETRAN\_LOG.LOG

where connector\_id is not null AND \_fivetran\_synced=:daterange

GROUP BY

connector\_id

order by ERROR\_RATE desc;



**2.6 Failed sync history**

select connector\_id,event,time\_stamp,parse\_json(message\_data):reason AS Error\_message

from FIVETRAN.FIVETRAN\_LOG.LOG

where (event='SEVERE' or event='WARNING') AND \_fivetran\_synced=:daterange and CONNECTOR\_ID=:Connector\_Id;

