**TN DB PATCHING INSTRUCTIONS**

**Pre-patch Validation Steps**:

1. Access Requirements:

* None as we assume DBA is running validation and has all requisite table accesses.
* TN Production database url is vacp0mmdb02tne.maxcorp.maximus; port 1521

1. Check if all jobs have completed processing – run the following query to check.

**SELECT \* FROM MAXDAT.CORP\_ETL\_JOB\_STATISTICS WHERE JOB\_STATUS\_CD = 'STARTED' AND (CURRENT\_DATE - JOB\_START\_DATE) < 1 ORDER BY JOB\_ID DESC;**

If the query returns no rows, then no jobs are running and we can proceed.

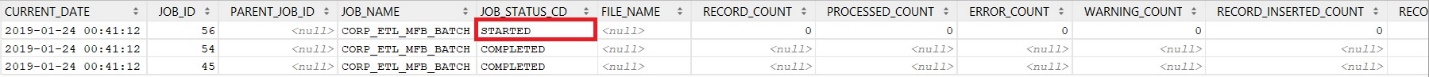
1. Work with the app admin to make sure all crons are turned off. The app admins are provided a pre-patching document with detailed steps on what to check/verify before turning the crons off.

**Post-patch validation Steps:**

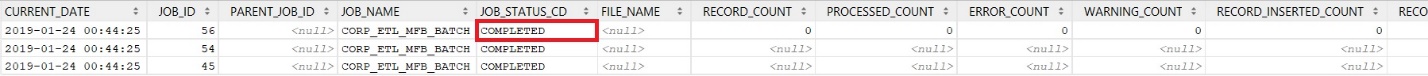
1. Work with app admin to ensure crons are turned back on after patching is completed.
2. After confirmation of crons being turned back on - Verify if the scheduled TN CORP\_ETL\_MFB\_BATCH (that runs at the **1st min** of the hour) is **running** and **completing** successfully as expected.

**SELECT** *CURRENT\_DATE*, **a**.*\** **FROM** MAXDAT.CORP\_ETL\_JOB\_STATISTICS **a WHERE JOB\_NAME** = **' CORP\_ETL\_MFB\_BATCH'  
ORDER BY JOB\_ID DESC**;

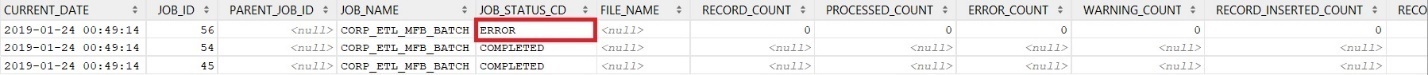
JOB\_STATUS\_CD should == **STARTED** when the job is **running**.



After the job successfully completes, JOB\_STATUS\_CD == **COMPLETED**



If CORP\_ETL\_MFB\_BATCH entries aren’t being created every 60 minutes, then the cron jobs are not turned on and app admin needs to be notified.



If CORP\_ETL\_MFB\_BATCH fails with entries of JOB\_STATUS\_CD = **ERROR** (jobs have completed but not successfully) – this needs to be looked into for any database errors.

1. Please reach out to App Admin to check TN Production logs for errors.

**Checking Queue Processor**

1. **Understanding Queue processor**

This happens after the ETL runs and is a process that batches the ETL changes and

prepares them to load into the semantic layer.

1. **Query to check Queue Processor jobs**

This query can be used to check what jobs are running (queue processor)

select \* from PROCESS\_BPM\_QUEUE\_JOB where STATUS != 'STOPPED' order by BSL\_ID asc, BDM\_ID asc, PBQJ\_ID asc;

Select

  --bsl.BSL\_ID,

  bsl.NAME "Staging Table Name",

--  count(bueq.BSL\_ID) "Total",

  sum(case when bueq.WROTE\_BPM\_SEMANTIC\_DATE is null then 1 else 0 end) "Semantic Pending",

  --sum( case when bueq.WROTE\_BPM\_SEMANTIC\_DATE is not null then 1 else 0 end) as Not\_Archived,

  nvl(round(((sysdate - min(EVENT\_DATE)) \* 24),2),0) "Hours Delay",

  nvl(to\_char(min(EVENT\_DATE),'YYYY-MM-DD HH24:MI'),'Current') "Oldest Unprocessed"

from BPM\_SOURCE\_LKUP bsl

left outer join BPM\_UPDATE\_EVENT\_QUEUE bueq on bsl.BSL\_ID = bueq.BSL\_ID

--and bsl.BSL\_ID = 15

group by bsl.BSL\_ID,NAME

order by bsl.BSL\_ID asc;

1. **Steps to follow when queue rows are stuck**

These are the steps to follow whenever queue rows are stuck and not processing.

1. Identify the processes that are stuck via the alert email

select \* from bpm\_source\_lkup – Identify the processes that need to be reset

1. Stop job processor:

execute MAXDAT\_ADMIN.SHUTDOWN\_JOBS;

2.Reset queue rows:

execute MAXDAT\_ADMIN.RESET\_BPM\_QUEUE\_ROWS\_BY\_BSL\_ID(BSL\_ID from the above query); --

After all the processes are reset one after the other

3. Restart job processor:

execute MAXDAT\_ADMIN.STARTUP\_JOBS;

Anyone who is available and has access to run MAXDAT\_ADMIN package in Production should be able to run the above. If no one is available or not responding, immediately create a jira, Emergency break fix CR and get it going. Pinging appadmin/DBA that you know will help speed up the process instead of waiting for the jira to be assigned to someone.