# Enterworks troubleshooting guide.

**1) Guide when reporting the slowness of using Enterworks UI or PROD UI was down, please making the following investigations**.

* 1. Try to Login to enable UI to check the system health widget; also try to login to the classic UI to check the system health widget and the data cache page.

Please take a look at the log files under Enterworks\logs\enable2020 folder, and Zip the log files under Enterworks\logs\enable2020 folder for further investigation.

* 1. Check Database Activities by DBA
* Check if there are long running sql query or blocking sql query
* Check if there are sleeping connections and the related sql queries
* Check if there are sleeping sessions  
  1. Check EnableServer Tomcat/JbossController/JbossWorkers
* Login to Tomcat Manager to check the following info, click ‘Server Status’ to open the Server Status page and right click to select ‘Save as’ to save it to tomcat1.html file; then click ‘Complete Server Status’

And then right click to select ‘Save as’ to save it as Tomcat2.html:

* + if there are long running http requests
  + If the current thread count reached the max thread count
  + If the JVM memory reached the max memory setting

2) Check each Jboss folder jbossXXX\standalone\deployments to see if it is still deployed; Check Tomcat bin folder to see if there are JVM’s memory dump files.

3) Perform the JVM thread dump for the EnableServer Tomcat/JbossController/JbossWorkers before restart the EPIM service using the jstack command:  
 You can find the Tomcat/Jboss PIDs from Windows TaskManager, and please make sure each box has Windows dll file sawindbg.dll installed in the D:\Enterworks\jdk\jre\bin folder.

* Open Windows Command Line in Administrator mode
* cd /d D:\Enterworks\jdk\bin
* Execute Jtsack command:

- jstack -F EPIM\_Tomcat\_PID > EPIMTomcatThreadDump1.log  
(for example: jstack -F 123456 > EPIMTomcatThreadDump1.log  
  
- jstack -F EPIM\_JBOSS\_CONTROLLER\_PID > EPIMJbossControllerThreadDump1.log  
(for example: jstack -F 78900 > EPIMJbossControllerThreadDump1.log  
  
- jstack -F EPIM\_JBOSS\_WORKER1\_PID > EPIMJbossWorker1ThreadDump1.log  
(for example: jstack -F 78901 > EPIMJbossWorker1ThreadDump1.log  
- jstack -F EPIM\_JBOSS\_WORKER2\_PID > EPIMJbossWorker2ThreadDump1.log  
(for example: jstack -F 78902 > EPIMJbossWorker2ThreadDump1.log  
  
- jstack -F EPIM\_JBOSS\_WORKER3\_PID > EPIMJbossWorker3ThreadDump1.log  
(for example: jstack -F 78903 > EPIMJbossWorker3ThreadDump1.log  
  
Collect the above thread dump log files, and send them to the Escalation Team for further information.

1.4 Zip all the log files under Enterworks\logs\EnbaleServer folder for each box and send it to the Escalation Team for further information.

1.5 Collect the file Enterworks\EnablseServer\tomcat\webapps\webcm\WEB-INF\classes\hibernate.properties

**2) Reporting the slowness of job processing or stuck jobs, please making the following investigations.**

* 1. The EPIM jobs can be initiated from the following two ways:
     1. Manually by users from UI, and those jobs show up in the Job Monitor.
     2. Automatically by the configured scheduled jobs via Service Framework, those jobs can be configured as normal EPIM jobs showing up in the Job Monitor or as custom SQL view jobs not showing up in the Job Monitor, and only showing up in its scheduled job repositories.
  2. If a stuck job is displayed in the job monitor page, check the following info:
     1. Check the job’s log file to see if any error occurred and if any batches took too long (could be related to DB sql queries blocking or deadlock issue, ask DBA to help to find out those kind of queries); also find out from the job monitor which worker was processing that job, and asking for that worker’s app log file and wildfly log file to do further investigation.
     2. If this stuck job is a system job, all the queued jobs will be waiting for this job to be finished, then they can be started to be processed.
     3. If this stuck job is non-system job, check the queued jobs to see if any system jobs are queued. Those system jobs will be started to be processed as system exclusive job when there is no other jobs processing.
     4. For the stuck jobs, please collect the following information before restart services:
        1. Download the stuck job’s log file from the Job Monitor for further investigation.
        2. find out the worker from the job monitor, then get that worker’s app log file and the wildfly log file, and job controller’s app log file and the wildfly log file for further investigation.
        3. Get the thread dump files on those workers, the controller and Tomcat.
        4. For the stuck job, find its job records from B\_JOB table and B\_JOB\_HISTORY table for the further investigation.
        5. Ask DBA to check if there are long running sql queries or blocking sql queries or deadlock sql quieres.
  3. If a stuck job is not displayed in the job monitor page, but it is displayed in a scheduled job repository, please check the following info:
     1. Open that scheduled job repository, check the status/log messages.
     2. Open the DesignConsole and monitor that scheduled flow to see where and why that scheduled job stuck, then locate the stuck work items and open them to check the errors and properties. Based on that information, search the EPX web log file, and the Bic log file, and Jboss log file for more detailed error information.
     3. Also check if that job stuck at its custom SQL query by asking DBA’s help.
     4. Also ask DBA to run the following EPX monitor sql query to see if there are big number of work items stack at some bic activities. If there are some BIC actives having large number of work item versions, then using the EPX DesignConsole to exam those stuck work item versions, and to decide if those stuck work items need to be purged.

select wis.ActivityType, count(wis.WORK\_VERSION\_ID) as NumWorkItems from (

SELECT case when (a.ACTIVITY\_TYPE\_CODE = 1) then 'AUTOMATIC: ' + a.ARC\_ACTOR\_NAME

when (a.ACTIVITY\_TYPE\_CODE = 2) then 'SUBFLOW'

when (a.ACTIVITY\_TYPE\_CODE = 3) then 'DECISION\_POINT'

when (a.ACTIVITY\_TYPE\_CODE = 4) then 'DISTRIBUTED\_SUBFLOW'

when (a.ACTIVITY\_TYPE\_CODE = 5) then 'WORK\_ITEM\_MERGE'

when (a.ACTIVITY\_TYPE\_CODE = 6) then 'ITERATION'

when (a.ACTIVITY\_TYPE\_CODE = 7) then 'JOIN: ' + case when (a.JOIN\_TYPE = 1) then 'OR' else 'AND' end

when (a.ACTIVITY\_TYPE\_CODE = 8) then 'MANUAL: ' + a.ARC\_ACTOR\_NAME

when (a.ACTIVITY\_TYPE\_CODE = 9) then 'ANONYMOUS'

when (a.ACTIVITY\_TYPE\_CODE = 10) then 'PERSONAL\_SUBFLOW'

when (a.ACTIVITY\_TYPE\_CODE = 11) then 'SPLIT'

when (a.ACTIVITY\_TYPE\_CODE = 12) then 'WORK\_ITEM\_REPEATER'

when (a.ACTIVITY\_TYPE\_CODE = 14) then 'WORK\_ITEM\_PURGE'

when (a.ACTIVITY\_TYPE\_CODE = 16) then 'ENDING\_POINT'

when (a.ACTIVITY\_TYPE\_CODE = 17) then 'LOAD\_BALANCE'

when (a.ACTIVITY\_TYPE\_CODE = 18) then 'CHANGE\_PRIORITY'

when (a.ACTIVITY\_TYPE\_CODE = 19) then 'SUBFLOW\_EXIT'

else cast(a.ACTIVITY\_TYPE\_CODE as VARCHAR)

end as ActivityType, wiv.WORK\_VERSION\_ID

FROM P\_PROCESS p with (NOLOCK)

JOIN P\_WORK\_ITEM wi with (NOLOCK) on p.PROCESS\_ID = wi.PROCESS\_ID

JOIN P\_WORK\_ITEM\_VERSION wiv with (NOLOCK) on wi.WORK\_ITEM\_ID = wiv.WORK\_ITEM\_ID

JOIN P\_ACTIVITY a with (NOLOCK) on a.ACTIVITY\_ID = wiv.ACTIVITY\_ID

WHERE wiv.DELETED\_IND <> 1

AND wiv.CURRENT\_IND = 1) wis

GROUP BY wis.ActivityType

order by wis.ActivityType

* 1. Try to abort and delete that stuck job to see if the queued jobs get started to be processed.