

Unable to USE Duplo in Production Env - Slow or Freezed UI

Incident Description:

Users were unable to use the Duplo System and UI was very unresponsive

How was the incident detected?

Users across regions reported sluggish performance of Duplo UI

What were the symptoms?

Users uploaded Bulk Import though UI more than 5k WO. Soon after that Duplo went into sluggish response.

What was the customer or business impact?

Users were not able to use Duplo software due to very sluggish response and any updates were not happening.

Duplo was not usable for a period of ~25mins.

How was the issue resolved?

The short term fix:

Investigation Part 1: Users uploaded Bulk Import though UI more than 5k WO. This triggered huge events within duplo and the DB CPU usage went till 99%.

Short term fix: was to double the DB Instance from 2->8vCPU and 16->64GB RAM

Investigation Part 2: With the higher DB instance, Duplo was loaded with Airing events of 3.9K. The DB CPU didnt spike this time with 3.9K events.

Investigation Part 3: Load Duplo with 10K events. This time we saw Duplo pods going into OOM Killed error.

Fix: Increase POD capabilities and count to handle more workload

Investigation Part 4: With Increased DB and POD instances, we loaded Duplo with 10K or higher Airing events workload.

This time DB CPU usage went to 99%, but handled the heavy load and came down to normal limits without any pods going down.

Observation and Conclusion: Duplo will be able to handle between 2k-3k airing events (new Bulk WO creations) with the current configurations.

The latest configurations are available here - [Duplo K8s Resource configuration.xlsx](#)

The long term fix: We have identified various areas for improvement:

1. [MSC-63331](#) - Optimize Duplo DB Queries
 - a. Move frequently called HQL queries to Native SQL Queries
 - b. Fetch only required columns from frequently called queries.

- c. Remove unnecessary calls to DB if any.
 2. [MSC-63324](#) - Move Read queries to read instances for Equal Service
 3. [MSC-63325](#) - Decrease DB Connections timeout
 4. [MSC-63326](#) - Remove multiple calls to same APIs from UI during initial load of the Dashboard
 5. [Slowness while updating orders tasks.docx](#)
-

Timeline

March 07 , 5:13 PM issue raised by users in Slack

March 07 , 5:15 PM Acknowledged by Engg team

March 07 , 8:40 PM DB instance capacity increased and unblocked Users

March 08, 8:21 PM Users again flagged as they Imported more WO

March 10, 4:41 PM PODs and Memory upgraded, system ready for use

5-Whys & Root-Cause

Problem Statement: Users were unable to use the Duplo System and UI was very unresponsive

Why did user face this issue? Users were importing WorkOrders in Bulk hence added more events into the system which caused the Database CPU to Spike and UI became unresponsive.

Why did we not face this issue before / Database load not tested ? Due to Max EMEA launch, lot of legacy WO were added to Duplo system for tracking purpose, this caused the spike. Normal user scenarios after launch will not have this workload.

Why Database couldnot handle the load? Database was running on a lower instance 2 CPU and 16GB RAM instance in production, which couldnot handle loads more than 300 events.

Why was DB instance not profiled for load earlier? Earlier load testing was done with less number of Airing events, 200 –300 spread across 1 hr.

Why was the load Testing lesser earlier? We couldn't predict that there will be more than 300 WO imports in an hour. Because of EMEA MAX launch, users had to add all the previous Show/Season in bulk.

Overall learnings and recommendations

1. Lesson Learned: Understand the total load before and after Launch.
 2. What went well?: Collaboration with Users, Engineering Team and Platform/DevOps teams
 3. What can be improved?: Educate Users about the load handling and maximum bulk imports. More than 50 imports, we recommend to use Excel Bulk Import than manual Bulk import. Manual Bulk Import was not supposed to handle more than 50 at a time.
-

Operational Excellence Learnings and recommendations

1. Load testing to cater to max WO, before launch. After launch we can reduce the same.
 2. Instantaneous alerts when DB Spiking to more than 99%, staying for more than a minute
-