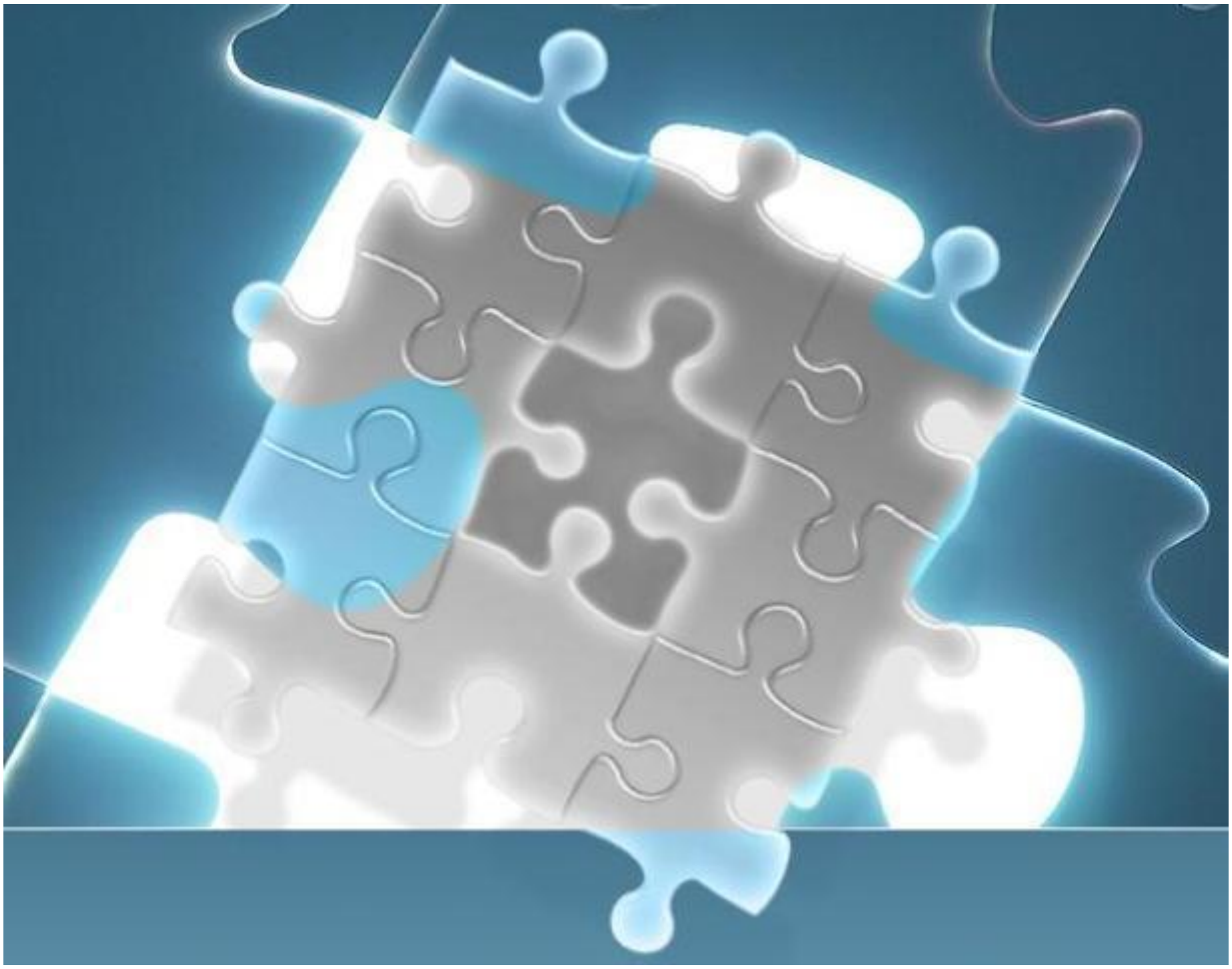
License Asset Manager™

Application Note

LAMUM – UM Performance Considerations



The following describes the ideal environment and resources for “top” performance, and issues that can affect response speed on the UM side of the tool. We address the issues that impact performance and give recommendations for optimizing UM response speed.

The “speed” we are referring to is;

1. Time to bring up a Usage Tab page.
2. Time to create a usage report.
3. Time to generate a report from a “Favorite” click.

LAMUM is Comprised of Two Elements- LAM and UM

LAM is the asset management side and addresses the License, Vendor and Tool data. It is static and runs very fast, because it accesses a static database and shows information there. Performance there should always be very good, and fast.

UM is the usage reporting side of the tool and is dynamic. It is constantly monitoring license servers (issuing lmstat), parsing log files, building tables, generating reports, monitoring alert settings, mailing alerts, etc. There is a lot going on in real time on the UM side of the tool. Many things can affect the UM side of the tool:

1. Number of daemons being monitored
2. Monitoring of non-Flex applications and parsing for Denials
3. Data collection interval setting
4. Size of the database, i.e. amount of legacy data to parse through
5. Size, speed and demand on the CPU and memory
6. Database accessibility/Disk speed
7. Network speed (including virus protection and firewalls)
8. DNS vs. IP setting in config.UM
9. Server vs. VM
10. # of Users (engineers being monitored)

Let’s Examine Each and Discuss How it May Affect UM performance

1. **Number of daemons**
   As the number of daemons being checked increases, UM performance could decline. You should not see any performance degradation with < 25 daemons. If the network speed is relatively high, and the cron interval in relatively large, you may not have any decline in performance. Beyond 25 daemons you may want to check network speed and set the Data Collection interval above 10 minutes. If you are monitoring > 50 daemons, please insure highest possible network speed and set Data collection interval 15 minutes or higher. As a general rule, each lmstat should be < 2 seconds per daemon.

2. **Monitoring of non-Flex applications and parsing for Denials**
   Monitoring non-Flex applications generally requires linking to and parsing remote log files. Examples might be Sentinel, LUM, Altium, Windchill, etc. FlexLM Denials reporting also requires parsing of remote Flex Debug Logs. In these cases, you are required provide a path to the logs and pick a time of day to collect and parse the data. Doing this during high network traffic periods, or periods of high LAMUM CPU/memory demand, can slow the process, and impact UM performance. Therefore, we suggest setting the log access and parsing to occur in off-hour periods (middle of the night?).

3. Data collection interval setting

As mentioned above, the cron interval set is critical to UM performance and data resolution. The more crons per hour, the better the data resolution will be, but the more impact on UM performance. Example: if you are monitoring 50 daemons every 5 minutes, on a slow network, it is possible that UM would be in the data collection mode constantly, and that could degrade performance. Even though UM can still be used to generate reports during data collection, it is still advisable to minimize the number of crons per hour, to a level that give you reasonably good data resolution and meaningful usage reports. A good rule is: 10 minutes for up to 25 daemons, 15 minutes for 25-50 daemons, 20+ minutes if over 50 daemons.

4. # of Users (engineers) being monitored

LAMUM can be affected when the number of Users gets very large (above 1,000). The lmstats can take longer. The writing and reading of User specific data can take longer. Report generation can take longer as more and more Users need to be incorporated into the report. There is not a whole lot that can be done to improve performance when dealing with large number of Users – keep data collection intervals longer, use a more powerful processor, create as Batch reports, TeamEDA is always looking at ways to make the UM processes and database calls more efficient.

5. Size, speed and demand on the CPU and memory

UM is a single threaded application, so a single socket should suffice. Clearly more CPU power, increasing number of cores (minimum 4 cores), increasing memory (8BG should be plenty) can help to improve performance on UM compute intensive tasks, like report generation. Not all of UM is compute intensive. Other applications/tasks, sharing the same CPU/memory can impact UM performance. That will impact UM performance significantly if resources are not available when UM needs them. It is wise to dedicate a CPU and memory to LAMUM. A dual-core medium speed processor, with 4 GB RAM should be enough.

6. Database size, access and disk speed

Since much of UM involves writing to and reading from the disk, size of the database and speed of the hard drive can affect UM performance. A SATA 7200rpm disk drive is a minimum. The faster the better. The other issue is around access to the database. Factors include a) the database should be located on the same server/VM as the LAMUM installation (avoid remote databases), b) there should be no gateways, barriers, overhead, or utilities that would impede UM writing to or reading from the database. There should only be one instance of MySQL on the LAMUM server. As the size of your database grows, UM performance may deteriorate. You should not see too much degradation until you get over 2GB. It will start to take longer for UM Tabs to come up, and for reports to generate. At 3GB it may take a minute for the Tabs to come up, and reports may take 2-3 minutes. The only solution is to decrease the database size, truncate old data, or split LAMUM. In version 6.4.2 there is a button to automatically truncate over 2 years of history. Consider turning that on.

7. Network speed (including virus protection and firewalls)

The single most important factor in UM performance is the Network speed. UM needs clear and fast access to and from the license servers. Any virus protection or firewall programs that change that access will impact UM performance. Moreover, the LAMUM server should made a trusted site with clean access to the network and license servers. If you are suffering UM performance degradation it would be wise to have a meeting with your IT department to ensure that network access to/from the license servers is clean, free of virus protection/firewall intrusion. Another issue is when/if a license server cannot be found or is not responding (server/daemon down). UM is programmed to try to get an lmstat for a specific amount of time, then it will "time out". If a license server is not responding in sequence, it will delay the next lmstat call, etc. This could change UM performance if waiting for returns. UM will keep trying until the lmstat is received or a "timeout" occurs, whichever comes first. All you can do is insure that license servers and daemon stay up.

8. **DNS vs. IP setting in config.php file**
   When installing UM, the license key includes a specific IP address, which you provided. That IP address must be static. If it is dynamic or changes, the LAMUM license key will not work. UM looks for the license key IP and compares it to the local IP every cron run. If the match, UM continues. Again, to minimize license checks, increase the cron interval. The IP address is stored in the config.php file. It is possible to change the config.php file to use a DNS or some other identifier. That is not advisable. That will slow down the License check process. UM works best when left as the IP address.

9. **Server vs. VM**
   LAMUM will run on a dedicated server or a server VM. The VM could be problematic if it introduces delays, barriers, or controls to the network or database. If you are using a VM, make sure there is nothing there that could change UM performance according the above, especially around network or database access.

10. **Number of LAMUM Users**
    Each click of a UM Tab, or a UM report creates a momentary surge on the CPU. A surge could be 40-50% of CPU capacity. The question is how many people would be using UM **at the same time**, clicking on UM Tabs or UM reports? If only 1 person, 2 cores will be enough. If 2-3 people, 4 cores should be fine. If 4+ go to 8 cores.

In short, some ways to make UM faster is to:

1. Increase dedicated VM compute resources
   a. Go to single socket, 4 cores or 8 cores
2. Eliminate VM and go to a physical server
3. Decrease database size
   a. Truncate over 2 years?
   b. Split LAMUM instances?
4. Decrease # daemons being monitored
5. Remove any "status=red" daemons
6. Decrease # of Users using UM at the same time.

UM Performance can start to degrade when database reaches 5GB. Could be sooner. To figure out your database size, go the LAMUM Backup folder and look at the most recent Backfile file. What is the size of the Backup? If more than 5GB consider action to reduce the database size.

Reducing the Database size

There are 2 ways to reduce the LAMUM database:

1. Remove the zeros
2. Truncate/archive old data

Removing the zeros

This procedure will reduce the database approx. 65%. That is significant! To remove the zeros, use the database reduction procedure. Go the Usage/Admin, **Database Reduction** and follow the instructions.

Truncate/archive old data

This procedure will reduce the database the % of time you are “truncating”. That can be significant if you are truncating/archiving 3-4 years of data! To truncate/archive old data go to Usage/Admin, **Archive Date & Location** and follow the instructions.

Managing your LAMUM database is a requirement. At some point, you will have to reduce its size. Please feel free to contact TeamEDA if you have any questions.

If you find something that you think will make the LAMUM performance better, please feel free to share that with us! Our usability for our customers is a top priority.