



JobsDB PERFORMANCE IMPROVEMENT WITH NetWorker 8.0



Gururaj Kulkarni
Principal Software QA Engineer
EMC Corporation
gururaj.kulkarni@emc.com

Soumya Gupta
Software QA Engineer
EMC Corporation
soumya.gupta@emc.com

Table of Contents

Introduction	2
Test Approach.....	3
Test Results	4
Purge Time.....	4
IOPS Requirement	6
Memory Utilization.....	8
CPU Utilization	9
Observations and Recommendations.....	10
Conclusion	10

Disclaimer: The views, processes, or methodologies published in this article are those of the authors. They do not necessarily reflect EMC Corporation's views, processes, or methodologies.

Introduction

EMC NetWorker® is an enterprise-class Backup and Recovery solution. It is three-tiered software:

1. NetWorker Server (co-ordinates the entire backup/recover process and tracks the metadata)
2. NetWorker Storage Node (connects to diverse storage devices and writes/reads data).
3. NetWorker Client (hosts the data to be backed up).

NetWorker Jobs data base is a key part in overall performance of NetWorker server which keeps job records information such as backup, recover, maintenance, and utility jobs. If not properly sized, growth of this jobsDB has an adverse impact on NetWorker server performance.

This Knowledge Sharing article provides a detailed update on JobsDB performance improvements made with NetWorker release 8.0., including a new JobsDB which uses SQLite as the backend database to store records. This article briefs the performance benefits with new JobsDB compared to earlier version such as NetWorker 7.6.3:

- CPU and memory used by nsrjobd in NetWorker 8.0 is close to 97% less, as compared to NetWorker 7.6.3.3 on a highly scaled JobsDB (1 million records).
- Purging is very efficient and is 1200% faster in NetWorker 8.0 as compared to NetWorker 7.6.3.3 on a highly scaled JobsDB (1 million records).

Test Approach

The test approach was:

- NetWorker server, storage nodes, and clients were installed with version 7.6.3.3 build 870.
- NetWorker server was installed on a LUN exported to the machine from a CLARiiON®.
- The JobsDB retention period was set to 30 days.
- An internal script was used to capture response time for the jobs daemon and query time, using the job query.

- 30000 records per day were created in the JobsDB. This included various types of job records such as backup jobs, clone jobs, recover jobs, and maintenance jobs. The JobsDB was scaled to 1,000,000 (1 million records).
- After every 100,000 records scaled in JobsDB, different job queries were run randomly. Also, whole records were queried for individual job types such as save job, session info, and so on. The IOPS and time taken to query the records were noted.
- Captured IOPS on disk where /nsr resides during JobsDB purge activities when the NetWorker server was idle.
- When the number of records in JobsDB is scaled to 1 million, the following jobdb related operations were completed:
 - Full Purging was completed on JobDB. The IOPS, the time taken to purge, and the number of records purged were captured.
 - Performed a fresh installation of the NetWorker 8.0 server with a new LUN, and executed test scenarios similar to NetWorker 7.6.3.3 and compiled the results. This ensured an accurate comparison between NetWorker 7.6.3.3 and NetWorker 8.0 JobsD.

Test Results

The Test results analysis are based on the following key objectives of JobsDB:

- Purge Time
- IOPS requirements
- Memory Utilization
- CPU Utilization

Purge Time

Purge time is the time it takes to purge the records in JobsDB. The retention period in JobsDB was set to a very high value to ensure that records were not purged during scalability tests. The nsrjobd starts the purge activity every hour. System resource utilization is captured during this purge activity. The purging time was measured in the following two ways:

- The time to purge the zero records in the JobsDB for every 100,000 records in JobsDB when the server is idle.
- The time to purge a certain number of records on a highly scaled JobsDB.

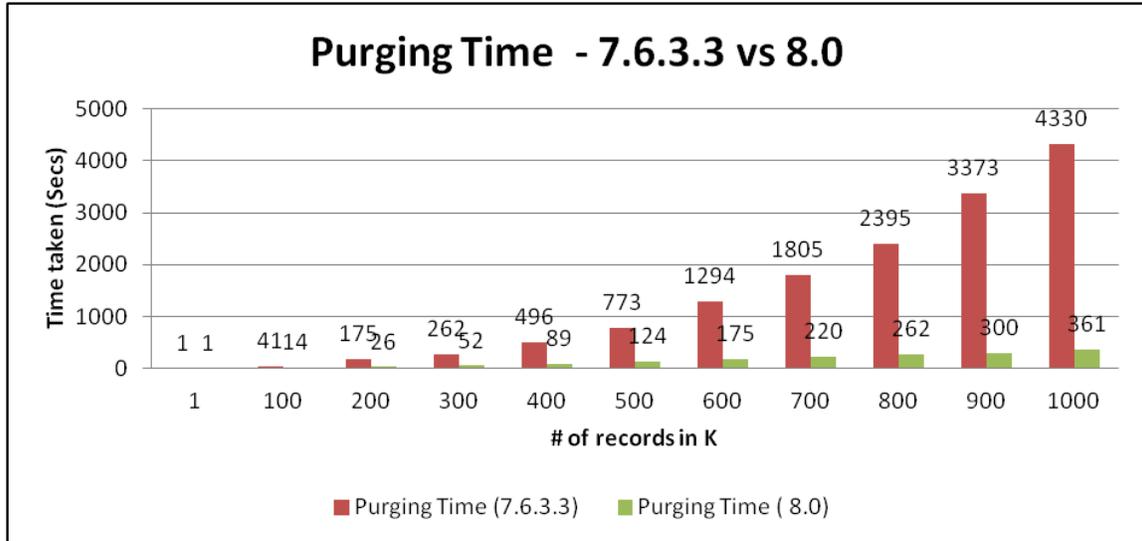


Figure 1: Time to purge zero records in JobsDB

Time taken to purge zero records in JobsDB at the regular increasing intervals of 100,000 records is illustrated in Figure 1. The time taken to purge the zero records in NetWorker 8.0 was minimal when compared to the same in NetWorker 7.6.3.3. When there were 1 million records in JobsDB, the purge time in NetWorker 8.0 was reduced by 1200% as compared to NetWorker 7.6.3.3.

JobsDB was scaled to 1 million records. The zero records were purged until the JobsDB scaled to 1 million records. The purging of the 1 million records was started by decreasing the “JobsDB retention in days” attribute from 30 days to 20,15,10,5,1 days respectively.

Figure 2 and Figure 3 describe the trend in the number of records purged (from 1 million records in JobsDB to only a few hundred records), and the time to purge the records in NetWorker 7.6.3.3 and NetWorker 8.0

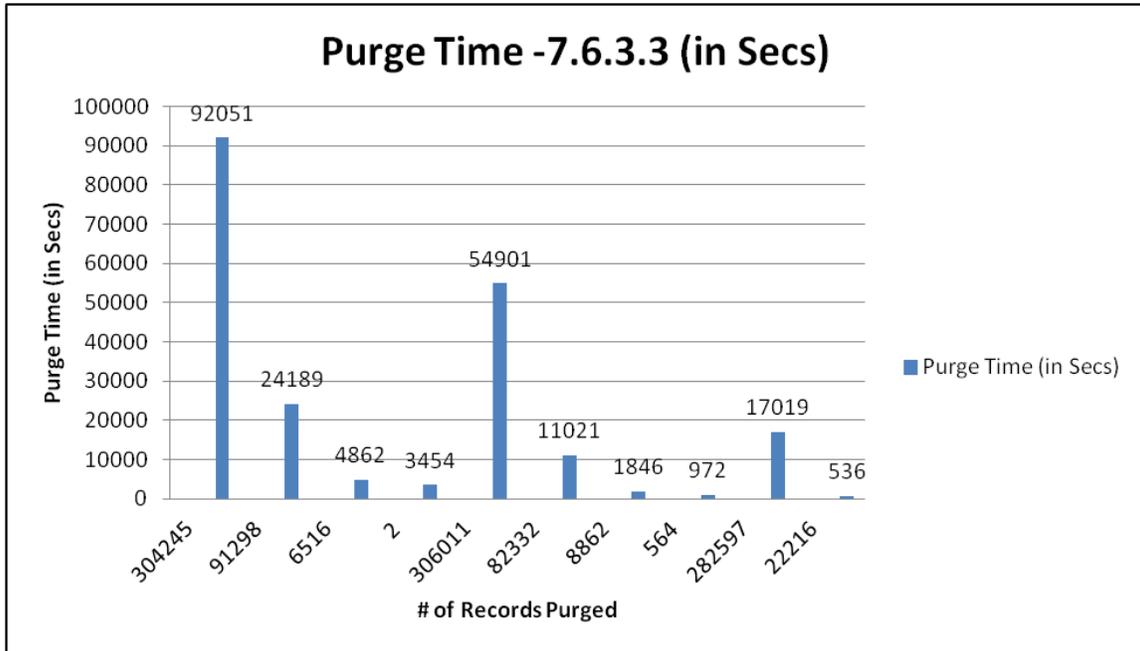


Figure 2: Time to purge certain number of records in 7.6.3.3 JobsDB

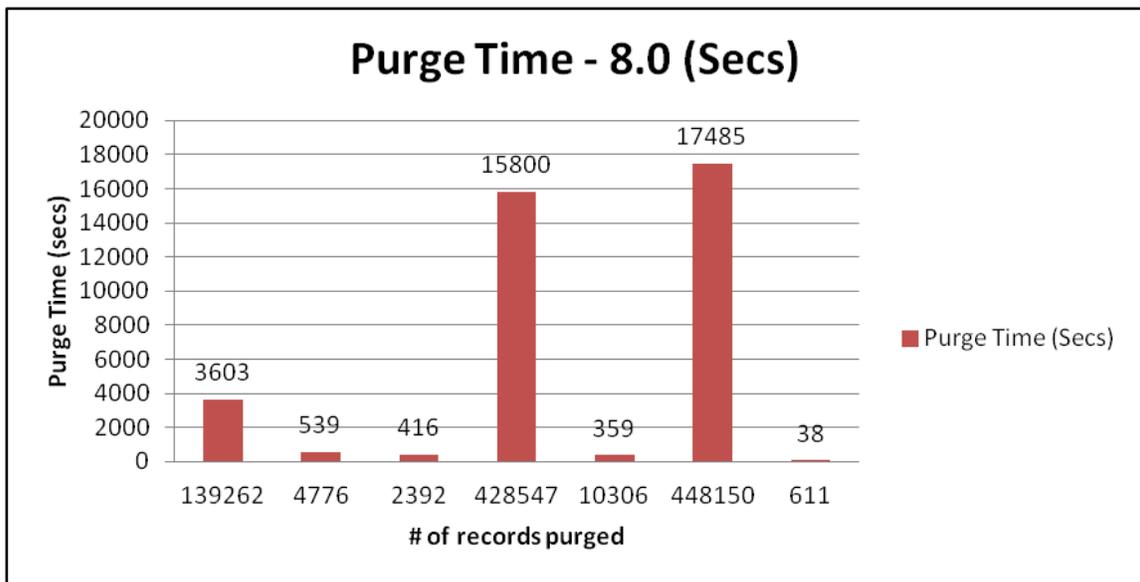


Figure 3: Time to purge certain number of records in 8.0 JobsDB

IOPS Requirement

IOPS is the number of input/output operations per second. During Jobs daemon performance tests, IOPS are measured on the disk where “/nsr” resides. The IOPS are captured during each purge activity. Figure 4 illustrates the IOPS measured for every 100,000 records scaled in JobsDB during purge activities.

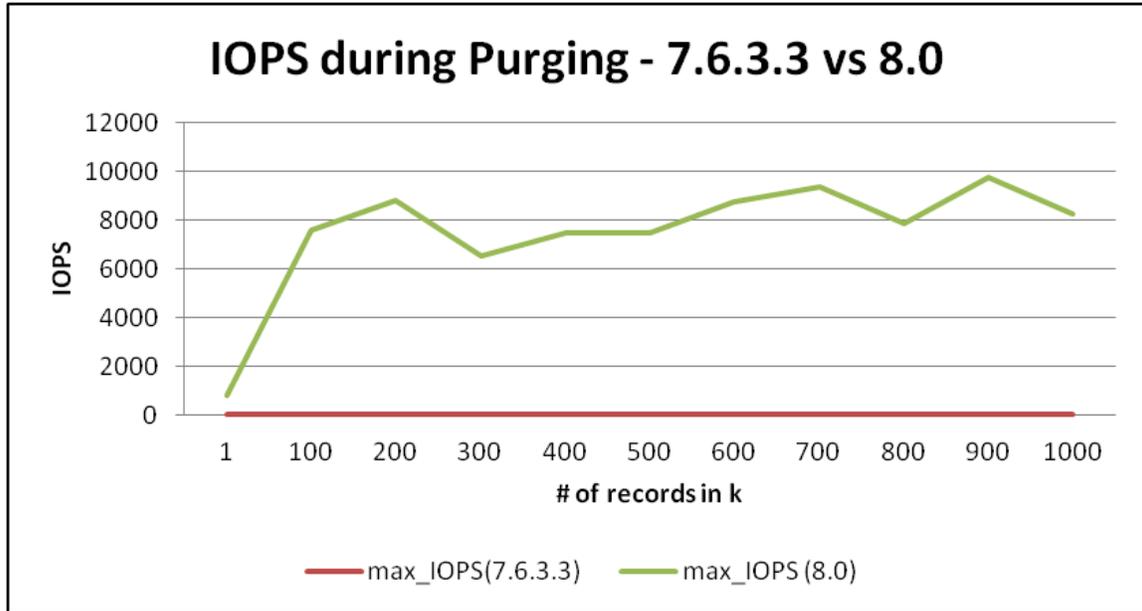


Figure 4: IOPS during the purge of zero records in JobsDB

The number of IOPS required in NetWorker 8.0 is more when compared to NetWorker 7.6.3.3. This is expected behavior with NetWorker 8.0 where concurrent I/Os occur at a time on the SQLite database during Vacuuming operations. However, in NetWorker 7.6.3.3 it is sequential with each I/O to the RAP database.

In NetWorker 8.0, the number of I/Os can increase to the level at which the I/O stack can provide. These are not mandatory IOPS during vacuuming operations. However, the operations will complete faster if the I/O stack can process as many I/Os as possible.

Figure 5 and Figure 6 illustrate the IOPS used to purge a number of records from JobsDB.

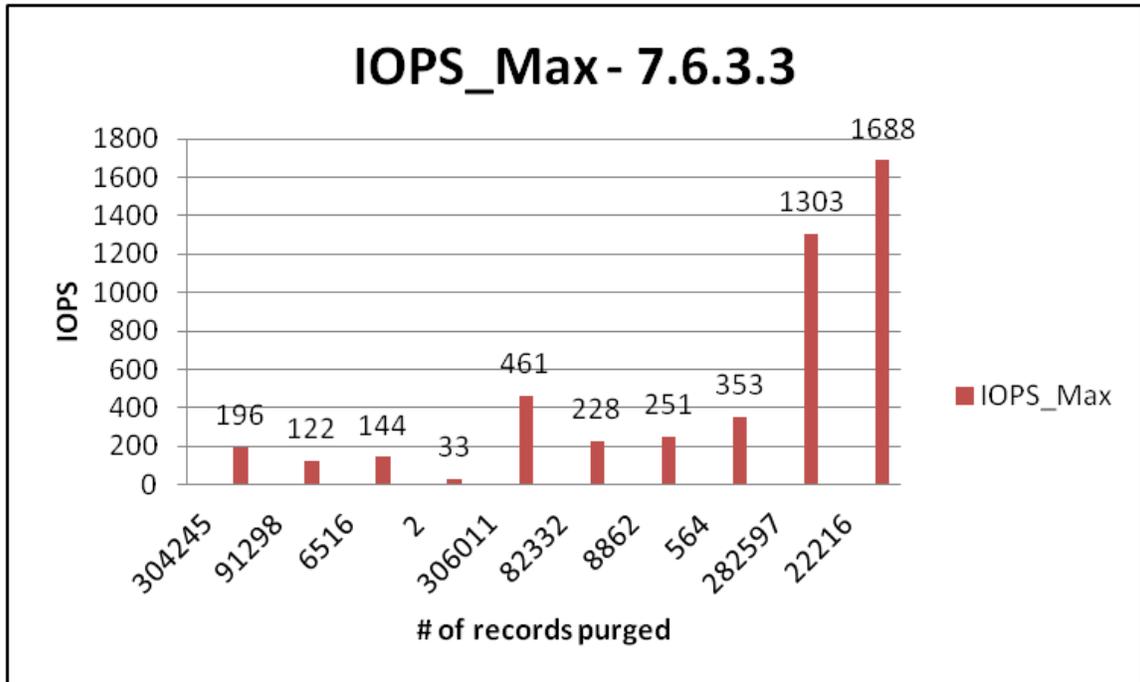


Figure 5: IOPS while purging a number of records in 7.6.3.3 JobsDB

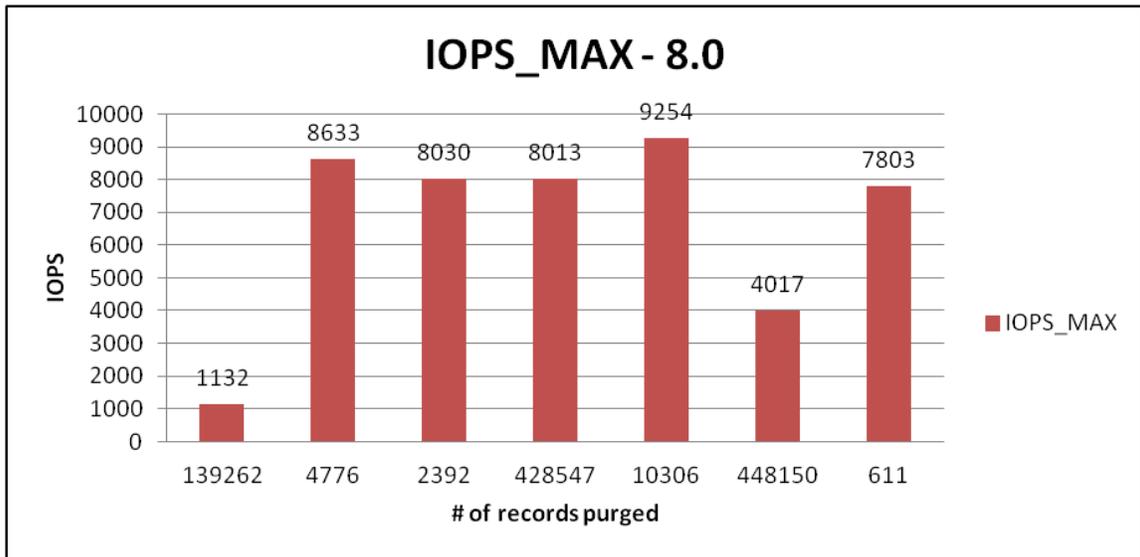


Figure 6: IOPS while purging a number of records in 8.0 JobsDB

Memory Utilization

Memory utilization is the amount of memory consumed by nsrjobd during the purge of zero records.

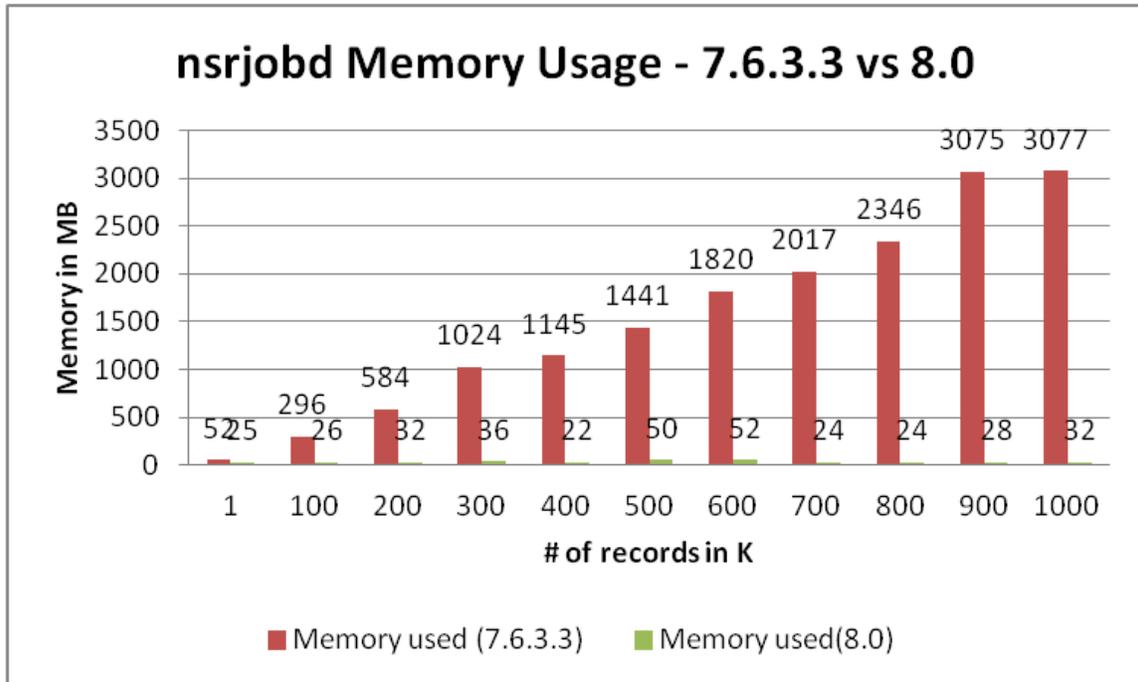


Figure 7: Memory use while purging zero records in JobsDB

The memory utilization by nsrjob daemon is detailed here. It is observed that memory utilization increases as the number of records increases (it reached 3.1 GB when the number of records in JobsDB was 1 million) in JobsDB in NetWorker 7.6.3.3. However, the memory utilization by nsrjob daemon is minimal in NetWorker 8.0. There is a reduction of 97% in memory use by nsrjobd in NetWorker 8.0 when compared to NetWorker 7.6.3.3.

CPU Utilization

The CPU utilization by the nsrjob daemon is detailed here. It was observed that system CPU utilization was more in NetWorker 7.6.3.3 as compared to NetWorker 8.0. The CPU usage by nsrjob daemon was reduced by 97% in NetWorker 8.0 as compared with NetWorker 7.6.3.3.

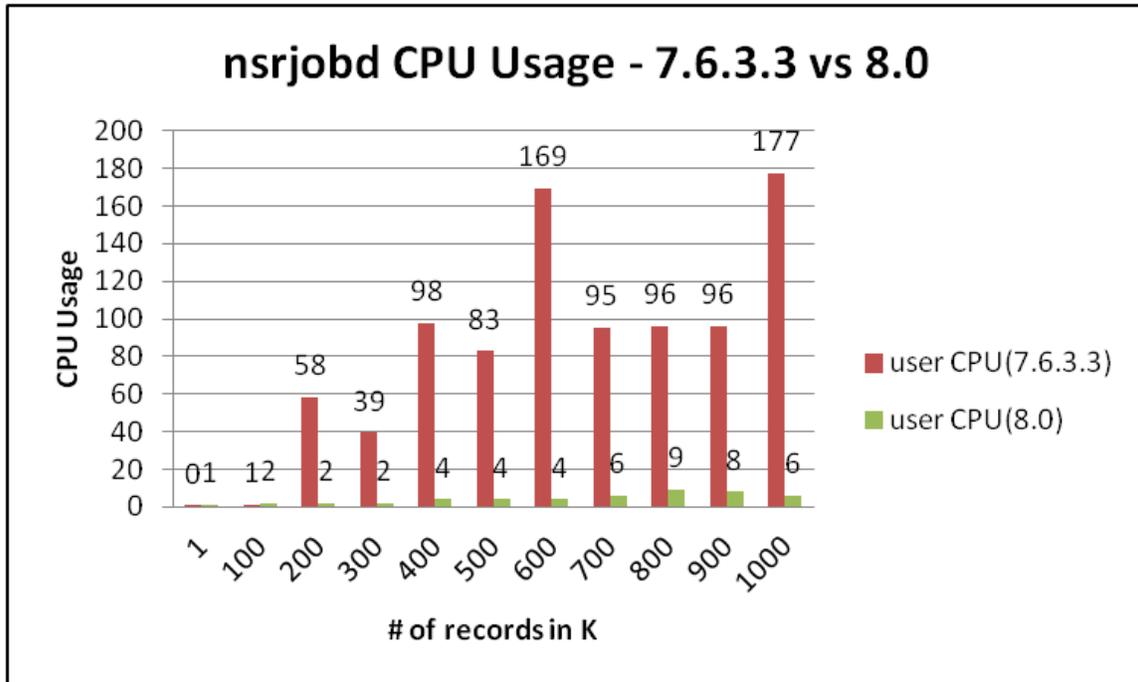


Figure 8: User CPU usage during the zero record purge in JobsDB

Observations and Recommendations

- It was observed that memory and CPU utilization by the nsrjob daemon in NetWorker 8.0 was reduced by almost 97% as compared with NetWorker 7.6.3.3 when the JobsDB was highly scaled.
- The time to purge zero records in NetWorker 8.0 on a highly scaled JobsDB decreased by 1200% as compared to the time to purge zero records on a highly scaled JobsDB in NetWorker 7.6.3.3. Also, the time to purge a large number of records on a highly scaled JobsDB in NetWorker 8.0 was less compared to that of NetWorker 7.6.3.3.

NOTE: It is recommended that you do not use jobquery on a highly scaled JobsDB as it will cause the nsrjobd memory to increase greatly. Also, all of the NetWorker operations will be affected until memory is released by nsrjobd.

Conclusion

The NetWorker 8.0 JobsDB on the SQLite database performs much better compared to the NetWorker 7.6.3.3 JobsDB that resides on the file system.

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS IS." EMC CORPORATION MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.