WHITE PAPER

Comparing the Performance of Oracle Database 12c on a Disk Array vs. the Fusion ioMemory PCIe Application Accelerator

SanDisk®

www.SanDisk.com

Table of Contents

Introduction

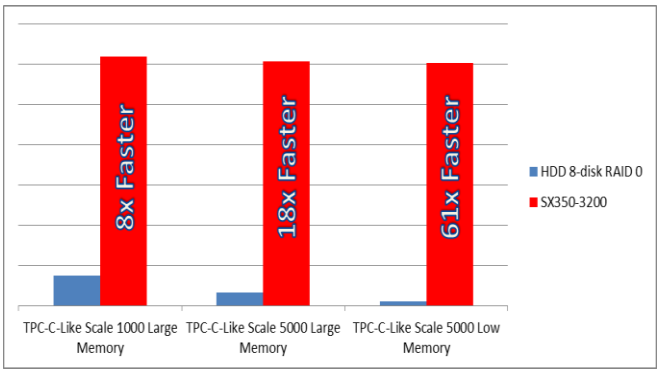
Dramatic improvements can be quickly and easily achieved in the application tier by accelerating the underlying storage layer. This paper examines the performance gains that can be achieved easily and repeatedly by moving an Oracle 12c database from a legacy array of hard disk drives to a modern Fusion ioMemory™ PCIe Application Accelerator from SanDisk. The target audience of this paper is Oracle database administrators and system architects, but the level of technical detail in this paper is appropriate for all audiences.

This paper is the first in a series that examines the performance gains realized by simply replacing the database's hard disks with a Fusion ioMemory storage device. The series topics are as follows:

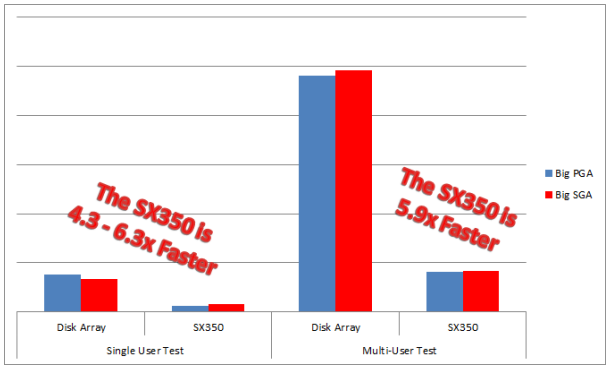
- Oracle 12c without new features or options (this paper)
- Oracle 12c new feature "Automatic Big Table Caching"
- Oracle 12c new feature "Force Full Database Caching"
- Oracle 12c new extra cost option "In-Memory Database Option"

Summary of Findings

Fusion ioMemory PCIe Application Accelerators from SanDisk accelerated the test workloads by 430% to 6100%, depending on the specific workload and other factors detailed in this paper.



OLTP Workload Relative Peformance



DSS Workload Relative Performance

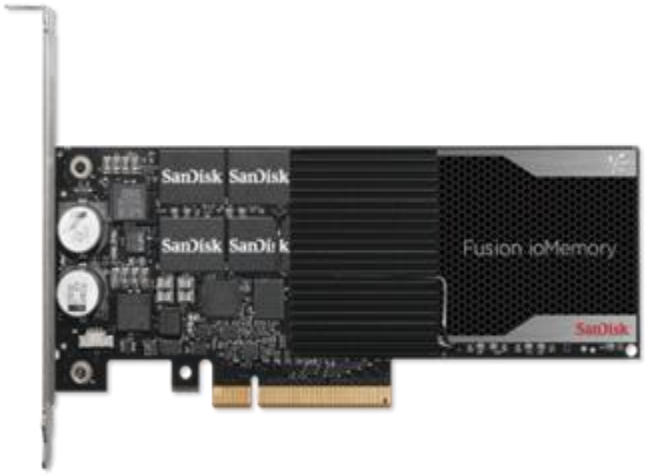
Databases typically have I/O bottlenecks that hurt overall application performance and worsen the user experience. If these bottlenecks are unresolved, customers may be driven to competitors. Organizations annually spend millions of dollars tweaking application code in hopes of improving performance by a few percentage points while ignoring the true problem. Resolving I/O bottlenecks through modernizing the storage layer is considerably faster, easier – and less expensive.

About the Fusion ioMemory PCIe Application Accelerator

Performance tests were conducted on a single Fusion ioMemory PCIe Application Accelerator model SX350-3200 by SanDisk. The SX350 line is Generation 3.5 of the Fusion ioMemory platform and features SanDisk NAND and an intelligent controller that supports all major operating systems, including Microsoft Windows, UNIX, Linux,

VMware ESXi, and Microsoft Hyper-V. The SX350 line is available in sizes ranging from 1300 to 6400 GB of addressable persistent flash memory. Fusion ioMemory PCIe Application Accelerators are also available in mezzanine form factors with capacities up to 1.6 TB for use in blade servers. Each device contains significantly more flash memory than is addressable, to enable wear leveling, longevity, and resiliency.

The test results in this paper are based on a single Fusion ioMemory SX350-3200 storage device, which has a raw usable capacity of 3.2 TB of SanDisk NAND flash. This device uses a low-profile PCIe 2.0 x8 slot, making it compatible with nearly all enterprise-class servers. Most servers support multiple devices, and each Fusion ioMemory SX350-3200 storage device keeps data center costs to a minimum by consuming less than 25 watts of power. The product's endurance rating is 11 petabytes written.



Performance of the SX350-3200 during our Oracle testing averaged 209,500 8KB random read IOPS with sub-millisecond latency and 2780 MB/s sequential reads.

Fusion ioMemory PCIe Application Accelerators are unique in their ability to sustain writes as well as or better than reads. Most Oracle databases perform more read operations than write operations, but writes can still be a bottleneck for Oracle. Consider that many OLTP and Operational Data Stores have a workload consisting of 40-50% writes. These include inserts, updates, and deletes to row data and indexes, and corresponding Redo and Undo generated by these operations. Even Decision Support Systems and analytics databases may experience slowness on checkpoints and logging. When selecting a storage product it is imperative to consider the database's dependency on write operations, not just read acceleration.

Additional information about the SX350 line of Fusion ioMemory PCIe Application Accelerators featured in this paper can be found in the video at https://www.youtube.com/watch?t=50&v=qweog75HTL8 and the data sheet at http://www.sandisk.com/assets/data-sheets/fusion-iomemory-sx350-pcie-application-accelerators-datasheet.pdf.

Our Test Methodology

Performance was measured for two Oracle 12c databases stored on an array of enterprise-grade hard disk drives, with RAID 0 used for maximum performance. Then, the two Oracle databases were moved to a single Fusion ioMemory SX350-3200 storage device and the tests repeated. All aspects of the test remained constant except for the underlying database storage: the server configuration, operating system, database configuration, etc., were not changed.

Red Hat Enterprise Linux 6.6 and Oracle 12cR1 were installed and configured per vendor documentation. ASM diskgroups and a test database were created as noted in Appendix A of this paper. HammerDB was installed and used to create a TPC-C-like schema of scale 5,000 for testing an OLTP workload, and a TPC-H-like schema of scale 300 for testing a Decision Support System (DSS) workload. Both schemas are approximately 400 GB. The OLTP schema grew during testing due to the heavy mix of transactions (table ORDER_LINES grew from 1.49 billion rows to 3.43 billion rows). To ensure fair testing against a growing schema, after every third run of a test the schema was dropped and restored from backup. The database instance was also restarted at that time so the first run of each test populated the cache, while the second and third runs of each test benefited from the cached data.

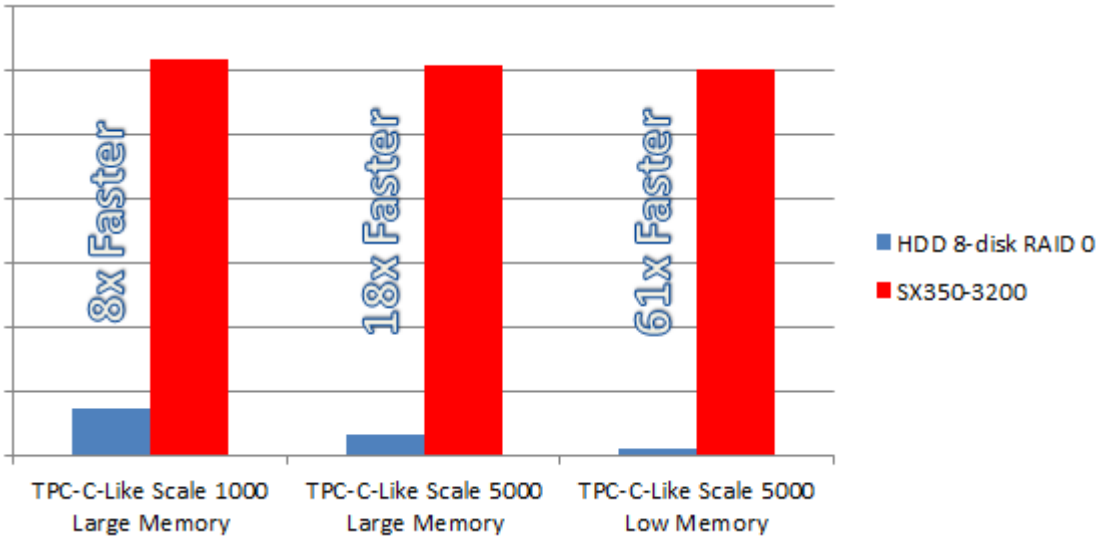
No database tuning was done to accommodate the Fusion ioMemory Platform. The same initialization parameter file was used for testing both classes of storage. Complete details of the test environment are provided in *Appendix A: Test Configuration*.

Test Results

The following sections review the results of performance testing with On-Line Transaction Processing (OLTP) and Decision Support System (DSS) workloads generated using the HammerDB application. Within HammerDB the workloads are referred to as being TPC-C-like and TPC-H-like. The HammerDB configuration is detailed in Appendix A.

OLTP Test Results

OLTP testing showed a single Fusion ioMemory PCIe Application Accelerator removed I/O bottlenecks and allowed Oracle to achieve performance gains of 8x to 61x when compared to the same database stored on an array of hard disks. While the actual performance numbers must be hidden per the Oracle End User License Agreement, the chart below shows that the Fusion ioMemory PCIe Application Accelerator maintains a consistently high level of performance under various database conditions while the array of disks provides neither high performance nor consistent performance across configurations.



The first configuration used a TPC-C-Like schema of scale 1,000. The Oracle buffer cache was sized twice as large as the data set, thus allowing Oracle to cache everything in memory. Oracle, by default, will not attempt to cache all data in memory even when adequate memory is available, so the database will always be I/O-dependent to some degree. Also, checkpoints and logging are I/O-dependent activities, regardless of the database cache size. We summarize the results as follows:

- The large DRAM configuration was 8X faster on Fusion ioMemory storage than on a disk array.
- Oracle on legacy disk storage does benefit, to some degree, from having more memory. For example, the high-memory configuration was 7x faster than the low-memory configuration.

The second configuration used five times as much data. The Oracle buffer cache was sized equal to the size of our test data set, thus allowing Oracle to cache all of the data, or as much data as Oracle determined to be appropriate. Performance of the database on legacy disk dropped considerably due to the additional reads and writes to disk (more data to read into the cache, and more data to flush on checkpoint). However, the same database stored on the Fusion ioMemory platform saw virtually no change in performance, due to its ability to sustain heavy reads and writes concurrently. The net result was that the Fusion ioMemory platform's advantage grew from 8X to 18X.

The third configuration perhaps best represents a typical production environment: the amount of usable system memory is often just a fraction of the amount of data in the Oracle database. In this test to simulate a realistic production configuration we sized the Oracle buffer cache to one-third the size of the data set. Performance on our test database on legacy disk fell sharply, while the Fusion ioMemory platform maintained its high performance. The net result was that the Fusion ioMemory platform allowed Oracle to complete 61 times as much work.

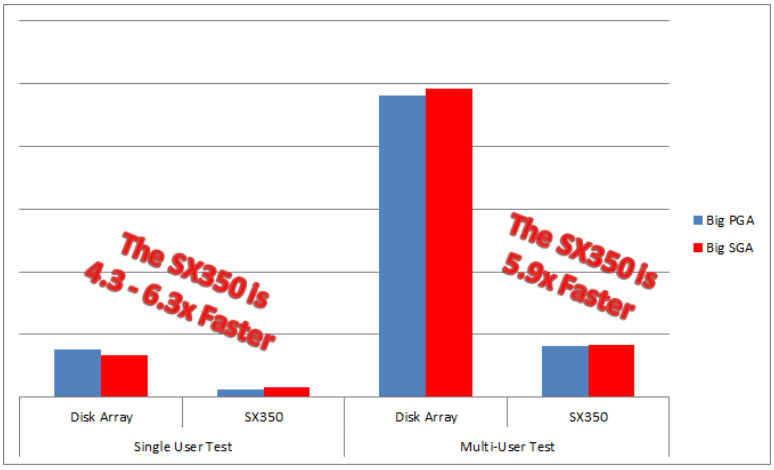
DSS Test Results

In this section we review the results of performance testing with a Decision Support System (DSS) workload instrumented using the HammerDB application as described in Appendix A of this paper. Actual performance metrics cannot be disclosed per the Oracle End User License Agreement, so the discussion is in relative terms such as "configuration A was 6 times faster than configuration B".

Shorter is better when it comes to graphing DSS test results! The DSS workload runs a set of 22 complex queries against billions of rows in a TPC-H-like schema. The workload is first run as a single user with parallelism, and then as multiple concurrent users with parallelism. HammerDB reports the time to complete each individual query, and the time to complete the entire set of queries. For brevity we examined the total run time only. Each configuration (or bar in the chart below), such as Single User Test with Disk Array Storage, was tested using four degrees of parallelism; each DOP was tested three times; and all 12 scores were averaged. This was repeated for each of the eight configurations for a total of 96 tests represented in the chart. Again, shorter run times are better.

The chart shows the relative run time of single and multi-user tests on each class of storage (disk and flash) and for each Oracle memory configuration (big PGA and big SGA). Multi-user tests take significantly longer than single-user tests, due to higher competition for system resources.

When the database is stored on a Fusion ioMemory SX350 device it outperforms the same database stored on an array of hard disks by 5.6x on average.



Observations

One of the most interesting observations came during DSS workload testing when we compared the effects of using a large PGA vs. a large SGA. Conventional wisdom suggests a large PGA is best for any workload with full table scans, because Oracle uses direct-path read to load data directly from storage into the PGA, bypassing the Oracle SGA completely. However, increasing the size of the SGA makes Oracle recalculate the threshold for "big" tables and considers more tables to be candidates for caching in the SGA. As a result, slightly fewer tables will use direct-path reads. The results were as follows:

- For single user tests with high parallelism, the bigger SGA sped up the hard disk-based database by 13%, but slowed down the flash-based database by 30%.
- For multi-user tests with low-to-moderate parallelism the big SGA and big PGA configurations performed equally on both classes of storage, to within 2%. Mathematically speaking, the large PGA did

provide better results in most cases. Realistically, the results were so close that no clear winner could be identified in the comparison of big PGA to big SGA.

Throwing large amounts of memory at an OLTP database workload running on legacy disk-based storage had a minor positive impact compared to moving the database to the Fusion ioMemory platform. This platform provided consistently high performance, regardless of having low or high amounts of DRAM in the server. By increasing the ratio of memory to data 15x – from a typical production case of 1:3 to an artificially high 5:1 – the performance of OLTP workloads on the hard disk-based system increased only 7x (less than a one-half gain per unit of extra memory), while moving the same database to the Fusion ioMemory platform with no additional SGA memory resulted in an impressive 61x performance increase.

Wait times for single-block random reads (e.g., the Oracle wait event db file sequential read) under an OLTP workload are noted in the below table. 144 total tests were run to measure latency: 72 for each class of storage, and within each class of storage 24 for each ratio of memory to data. Thus, each number in the table below represents the average wait times from 24 tests.

	Low Data, High Memory	High Data, High Memory	High Data, Low Memory
HDD Array	21.87 ms	45.99 ms	61.54 ms
SX350	0.65 ms	0.61 ms	0.46 ms

Log File Sync is another important metrics to be measured. The below table reflects the average wait times for this metric across each class of storage and each ratio of data to memory. 144 total tests were run: 72 for each class of storage, and within each class of storage 24 tests were run for each ratio of memory to data. Thus, each number in the below table represents the average wait times from 24 tests:

	Low Data, High Memory	High Data, High Memory	High Data, Low Memory
HDD Array	31.72 ms	65.32 ms	125.76 ms
SX350	1.56 ms	1.58 ms	1.50 ms

Conclusions

Databases run significantly faster when migrated from legacy disk storage to modern Fusion ioMemory PCIe Application Accelerators from SanDisk. When powered by the Fusion ioMemory Platform the DSS workload was accelerated 600%, and the OLTP workload was accelerated between 800% and 6100%, depending on several factors, compared to running the same workloads against the same databases stored on an array of hard disks.

For more information about the Fusion ioMemory PCIe Application Accelerators from SanDisk, please visit our landing page at http://www.sandisk.com/enterprise/pcie_flash/ or simply call 1-800-578-6007.

Appendix A: Test Configuration

Server Specs

The database server is 2-socket system. Physically, it is a 2U rack mount server. There are two 14-core processors with hyperthreading enabled, for a total of 28 cores and 56 threads. The processor model is Intel® Xeon® CPU E5-2697 v3 (Grantley) @ 2.60 GHz.

The server has 768 GB RAM. 600 GB is allocated to HugePages for use by the Oracle SGAs of all databases on the server. 100 GB is logically allocated to the Oracle PGA for all databases on the server. The remaining 68 GB supports the operating system, drivers, and the VSL™ (Virtual Storage Layer) software. Memory does not need to be kept in reserve for a filesystem page cache, because Oracle ASM is used with direct I/O – all reads go directly into either the Oracle SGA or PGA.

Storage consists of ten 1 TB hard disk drives and two Fusion ioMemory SX350-3200 devices. The Fusion ioMemory storage devices use VSL software version 4.2.1 and firmware version 8.9.1. Two of the hard disks are used for a mirrored boot drive, and the other eight disks are configured as a RAID 0 group managed by Oracle ASM for database files. Only one Fusion ioMemory storage device is used for database storage and for comparison to the array of disks. The second Fusion ioMemory storage device is used for scratch space and backups: storing backups on flash allows the database to be re-baselined very quickly between each test. All storage devices (hard disks and flash) were formatted with a sector size of 512 bytes. Partitions were created using the Linux “parted” utility. Ownership and permissions were set on the partitions using the Linux udev rules facility.

The eight hard disk drives and one Fusion ioMemory storage device used for database storage were left unformatted and placed into an Oracle ASM named DATA with External Redundancy. The second Fusion ioMemory storage device, which was only used for backups and scratch space, was formatted and mounted with the EXT4 file system.

Operating System

The operating system is Red Hat Enterprise Linux (RHEL) 6.6 with kernel 2.6.32-504.12.2.el6.x86_64. The Java version was upgraded to 1.8.0_45., and the following customizations were made to the OS environment:

- Kernel tuning parameters were adjusted based on Oracle documentation and best practices.
- The /etc/grub.conf file was updated to disable c-states and Transparent HugePages (THP).
- The /etc/sysconfig/cpuspeed file was edited to set the p-state to GOVERNOR=PERFORMANCE.
- The /dev/shm device was increased from its default size of ½ RAM to 605 GB.
- Linux HugePages were enabled by increasing the size from 0 to 307,200 pages (600 GB) in /etc/sysctl.conf.

To ensure the HugePages fit into the shared memory space without issue, we made the shared memory device slightly larger – 605 GB – by editing /etc/fstab and adjusting the /dev/shm device. By default, on Linux the device /dev/shm is sized to only one-half of DRAM (384 GB on our test server) and HugePages is disabled.

Test Software

The following test software was installed on the server:

- HammerDB version 2.16 for Linux (64-bit). This software was used to create the test schemas and to execute all tests. The schemas and data follow the general conventions of TPC-C and TPC-H, but the test procedures do not. For example, all tests were run directly on the database server rather than using test terminals.
- Flexible IO Tester (fio) version 2.2.7 was used to test each storage device prior to installing Oracle, just to make sure all devices were performing acceptably.
- i7z was used to test the CPUs for c-state issues, after both the BIOS and operating system were configured to disable c-states. The i7z utility reported all processors operating at c-state 0 as desired.

Oracle Database Software

The RDBMS is Oracle Database Server Enterprise Edition version 12.1.0.2 with Grid Infrastructure (ASM). The software was obtained from My Oracle Support using patchset ID number 17694377. Only .zip files 1-4 were downloaded and installed. Oracle automatically installs many optional (separately licensed) software products, but only the options listed below were used in our testing:

- Oracle Enterprise Manager Diagnostic and Tuning Packs, required for generating AWR reports
- Oracle Partitioning: Only table TPCC.ORDER_LINES was partitioned. No other tables or indexes in any application schemas were partitioned.

Oracle ASM Configuration

Oracle Automatic Storage Management (ASM) is used as the file system and logical volume manager. The server has one ASM instance named +ASM, and two ASM disk groups named GRID and DATA, as described below. All disk groups use default settings for sector size, allocation unit size, etc.

- The GRID disk group is created during installation of the Oracle Grid Infrastructure software. In non-RAC environments it provides a simple place holder for the ASM instance. It holds only the ASM instance's parameter file and password file, but in RAC environments it also holds CRS metadata. Our GRID disk group uses a 1 GB partition from all eight hard disks and the one flash device, and it is configured with High Redundancy.
- The DATA disk group uses the remaining capacity from each storage device and is configured with External Redundancy. The purpose of DATA is to hold all database files, including data files, redo logs, temp, etc. To start, the DATA disk group used all eight hard disk drives and none of the flash memory. In the second round of testing, one flash device was added and the hard disks removed, so that ASM moved the database from the old to the new storage. ASM automatically moves extents to new devices as they are added and removes extents from the old devices as they are ejected.

Oracle Database Configuration

Two databases named DB1 and DB2 were created using the Oracle Database Configuration Assistant (DBCA). Both databases share one ASM disk group named DATA. The first database (DB1) supports OLTP workloads: it uses an 8KB block size and does not use parallelism except during maintenance operations. The second database (DB2) supports DSS workloads: it uses a 32KB block size and parallel query processes.

All database files for both databases are stored on the ASM disk group "DATA". The control files and online redo logs are multiplexed, so there are two copies of each of these files written in parallel by Oracle. (This is necessary to protect the database, as it is stored on an ASM disk group without redundancy or underlying RAID protection.) The on-line redo logs are 5 GB per member, 2 members per group, and 5 groups per database.

All tablespaces are Bigfile Tablespaces and use the database's default block size. All system datafiles and tempfiles were created with an initial size of 1 GB with AUTOEXTEND ON NEXT 1G. Application tablespace datafiles are documented in the section on HammerDB Configuration.

HammerDB Configuration

Tablespaces for the HammerDB schemas were created in SQL*Plus using the default names suggested by HammerDB, which are TPCCTAB and TPCHTAB. Both tablespaces were created as a Bigfile Tablespaces of size 500 GB with attributes AUTOEXTEND ON NEXT 100G MAXSIZE 1T. Tablespace TPCCTAB was created in database DB1, and tablespace TPCHTAB was created in database DB2.

Schemas were created and populated using HammerDB. The schemas generally follow the guidelines of the TPC-C and TPC-H specification; however, no attempt was made to perform actual TPC benchmarking. The TPC-C-like schema was used for OLTP testing and only implemented in database DB1, and the TPC-H-Like schema was used for DSS testing and only implemented in database DB2.

HammerDB was installed directly on the database server in a directory under the oracle user's home. The full path is `/home/oracle/software/HammerDB-2.16`. The time to create and populate the two schemas was not recorded and is considered irrelevant to this paper. All tests were executed directly on the server to ensure network latency did not skew test results.

The scale, or size, of each schema is noted below:

- The scale of the TPC-C-Like schema is 5,000. This equates to approximately 275 GB of data and 130 GB of indexes, for a total schema size of 405 GB. This schema grows during testing, and so the scale 5,000 was appropriate for the SGA size. Table ORDER_LINE is partitioned by hash, but no other tables are partitioned, as this is the default behavior when partitioning is selected in the HammerDB application.
- The scale of the TPC-H-Like schema is 300. The largest table LINEITEM starts with 1.8 billion rows, which consumes 234 GB of space, not counting indexes. The total schema size is just under 400 GB, with 340 GB of data and 50 GB of indexes. Oracle Partitioning is not used, as it is not available in the HammerDB application for TPC-H-like schemas. Users are free to manually partition tables and indexes, but this was not done in our testing.

The multi-user DSS testing was performed with eight query streams and four different degrees of parallelism: 7, 9, 11, and 13. The minimum number of query streams is dictated by the scale of data, in following the spirit of the TPC-H specification, and in our case a data scale of 300 called for a minimum of 6 query streams. We used 8 query streams to account for the newer and more powerful processors and memory DIMMs found in today's servers. The degree of parallelism is left to the user; we chose levels that were appropriate given the server specifications.

DPL2015-01 EN 20170628