HPE Reference Architecture for Microsoft SQL Server 2017 on Linux with HPE Scalable Persistent Memory on HPE ProLiant DL380 Gen10

Reference Architecture

Contents

Executive summary ... 3
Solution overview ... 4
  HPE Scalable Persistent Memory ... 4
  Microsoft SQL Server 2017 on Linux ... 5
  SUSE Linux Enterprise Server ... 6
  HPE ProLiant DL380 Gen10 server ... 6
Solution components ... 7
  Hardware ... 8
Best practices and configuration guidance for the solution ... 9
  HPE ProLiant DL380 Gen10 server ... 9
  SQL Server Configuration guidance ... 9
  SQL Server Resource Governor ... 10
  SLES 12 SP3 kernel parameters ... 10
Capacity and sizing ... 11
  Workload description ... 11
  Analysis and recommendations ... 12
Summary ... 15
  Implementing a proof-of-concept ... 15
Appendix A: Bill of materials ... 16
Resources and additional links ... 17

Executive summary

With the introduction of Microsoft® SQL Server 2017 on Linux® along with HPE Scalable Persistent Memory on HPE ProLiant Gen10 servers, customers have an exciting new platform for deploying a diskless database solution with substantially improved performance and lower costs. SQL Server 2017 brings Microsoft's industry-leading relational database engine to the enterprise Linux ecosystem. HPE Scalable Persistent Memory runs at memory speeds with terabyte capacity, delivering outstanding performance to put data to work more quickly in your business. It is ideal for accelerating database and analytics workloads.

Hewlett Packard Enterprise published a world record TPC-H™ Benchmark @ 1000GB result[1] using Microsoft SQL Server 2017 on Linux with HPE Scalable Persistent Memory running on SUSE® Linux Enterprise Server on an HPE ProLiant DL380 Gen10 server. It was the first 1000GB TPC-H result with over 1 million queries per hour (QphH), with a price of only $0.47 USD per QphH@1000GB. Compared to an HPE ProLiant DL380 Gen9 with traditional SSD storage[2], this solution provides:

- Up to a 31% drop in hardware cost
- Up to 41% performance improvement with Data Analytics workloads
- Up to 85% reduction in rack space due to the removal of storage, which provides:
  - Significant power and cooling savings, or
  - Room to add more servers to that space for increased data center compute capacity

In addition, when using HPE Scalable Persistent Memory in place of traditional SSD storage on an otherwise similarly configured Gen10 server, individual queries completed as much as 22 times faster.

This solution is ideal for:

- Customers who are looking to improve cost/transaction on general purpose and Entry/Mid-level SQL Server database deployments.
- Customers running older versions of SQL Server on older infrastructure and seeking better transaction performance.
- Oracle customers who wish to stay on Linux and migrate to a lower cost solution.
- Customers who are looking to standardize their OS on Linux, coexisting with other database solutions.

Target audience: This Hewlett Packard Enterprise Reference Architecture white paper is designed for IT professionals who use, program, manage, or administer databases that require scalability, reliability, and performance. Specifically, this information is intended for those who evaluate, recommend, or design new IT high-performance architectures.

Document purpose: The purpose of this document is to describe a Reference Architecture, highlighting recognizable benefits to technical audiences.

[1] TPC-H results show the HPE ProLiant DL380 Gen10 with a result of 1,009,065.5 QphH@1000GB and $0.47 USD per QphH@1000GB with system availability as of 04/30/18. Results published 11/17/17, see www.tpc.org/3331. This is the top TPC-H Benchmark @1000GB result as of the publication of this paper, 02/07/18.
[2] TPC-H results show the HPE ProLiant DL380 Gen9 with a result of 717,101.3 QphH@1000GB and $0.61 USD per QphH@1000GB with system availability as of 10/19/17. Results published 4/17/17, see www.tpc.org/3327.

Solution overview

This solution is based upon Microsoft SQL Server 2017 on Linux and Hewlett Packard Enterprise Scalable Persistent Memory running on SUSE Linux Enterprise Server. This high-capacity persistent memory product allows customers to put an entire SQL Server database in persistent memory. The underlying hardware is the HPE ProLiant DL380 Gen10 server. An overview of each of these components is provided below.

HPE Scalable Persistent Memory

An integrated storage solution, HPE Scalable Persistent Memory runs at memory speeds with terabyte capacity using resilient technology currently deployed in your data centers. It includes a DRAM layer to accelerate applications, a dedicated flash tier for persistency, and backup power to facilitate moving data from DRAM to Flash. This enables a terabyte-scale storage solution running at DRAM speeds, eliminating storage bottlenecks that typically result from writing to slower storage media.

In developing persistent memory, Hewlett Packard Enterprise has worked side by side with the leading operating system and enterprise application providers to create a software ecosystem that synchronizes with the technology. This empowers businesses with fast and reliable performance for all their key business workloads.

HPE Persistent Memory will help you enhance your competitive edge, identify areas to cut costs, and uncover new ways to drive revenue. Hewlett Packard Enterprise broke ground in the industry with a massive persistent memory breakthrough and leads the industry with a server platform offering persistent memory in an NVDIMM form factor. Now, Hewlett Packard Enterprise introduces HPE Scalable Persistent Memory, a higher capacity NVDIMM and a new integrated solution that runs at memory speeds with terabyte scale capacity.

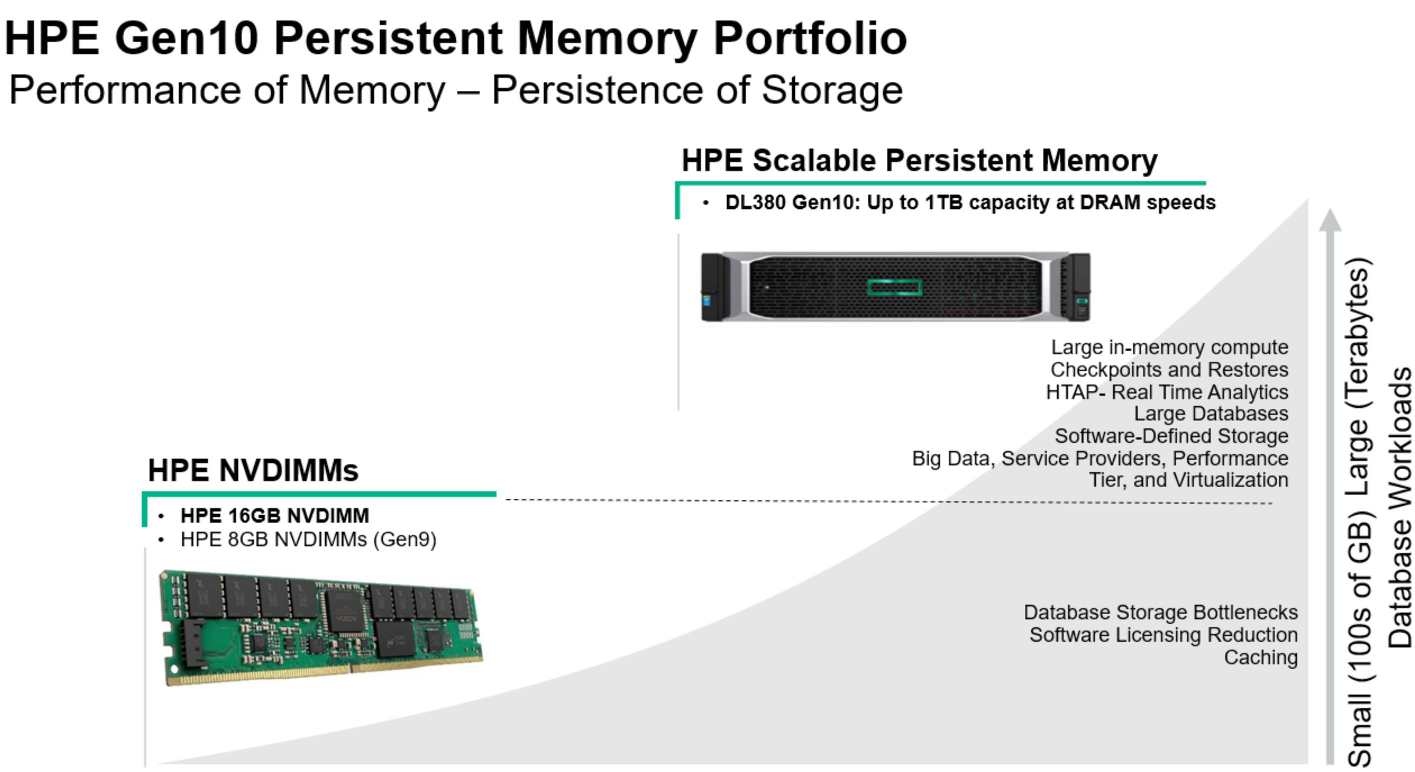

Figure 1. HPE Gen10 Persistent Memory Portfolio

Microsoft SQL Server 2017 on Linux

SQL Server 2017 brings the best features of the Microsoft relational database engine to the open source Linux operating system. Not a rewrite or a port, SQL Server 2017 on Linux is the same Microsoft relational database management system now available on more operating systems which is made possible by the SQL Server Platform Abstraction Layer (SQLPAL).

SQLPAL is responsible for abstraction of the calls and semantics of the underlying operating system and its libraries from the software itself, which enables the same subset of Windows® libraries used by SQL Server to run on Linux. All resources in the process are managed by SQLPAL. In this new architecture, even the Win32 and NT APIs are based on SQLPAL so every memory allocation and thread would be controlled by SQLPAL. This approach delivers the known reliability and performance of SQL Server to the Linux ecosystem[3].

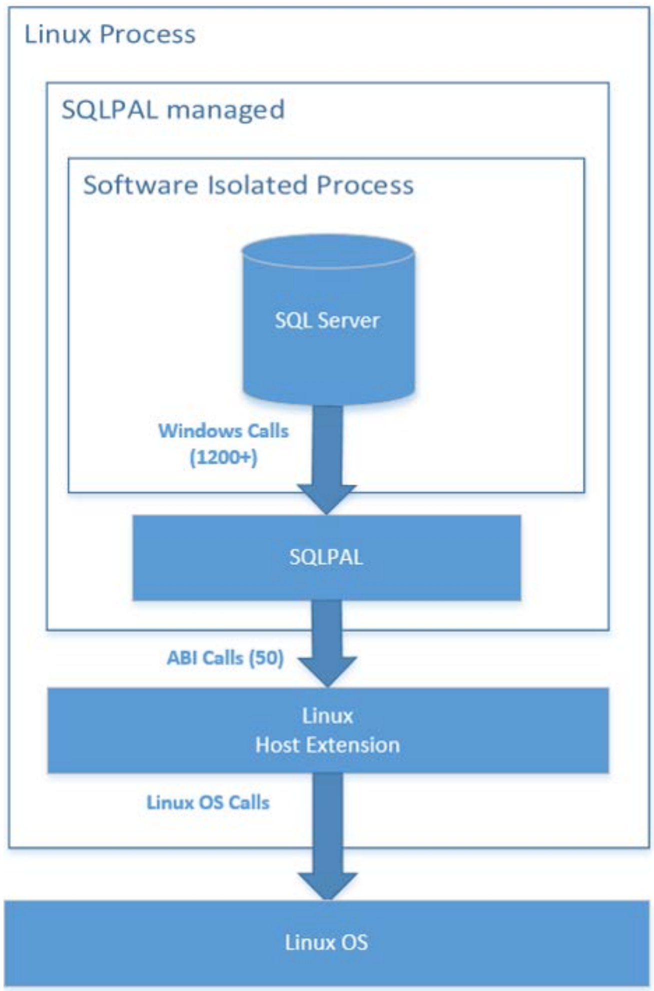

Figure 2. SQLPAL overview for SQL Server on Linux

Industry-leading performance features available on Linux

- Features like Real-Time Operational Analytics, in-memory OLTP, in-memory columnstore, compression, and partitioning are available on Linux.
- SQL Server 2017 includes many new Database Engine features and performance improvements, including resumable online index rebuild, automatic database tuning, graph database capability, and adaptive query processing to mention a few. To learn more about all features, capabilities, and services available on SQL Server 2017, see the Resources and additional links for Microsoft SQL Server on Linux.

[3] "Microsoft SQL Server 2017 on Linux Quick Start Guide", at http://info.microsoft.com/rs/157-GQE-382/images/EN-US-CNTNT-Whitepaper-DBMC-SQLServerOnLinuxQuickStart_Updated.pdf

Flexibility for high availability and disaster recovery on Linux

- Read-Scale Availability Groups provide load balancing of read-only workload replicas in mixed-OS environments but not high availability. This architecture does not require a cluster manager.
- Always On Failover Cluster Instance (FCI) provides instance-level high availability, allowing a SQL Server instance to remain available during planned or unplanned outages caused by hardware and software failures, or system maintenance.[4]
- Always On Availability Groups provide database-level high availability, disaster recovery, and read-scale balancing.

Security

- Security features such as Auditing, Transparent Data Encryption, Row-Level Security, Always Encrypted, and Dynamic Data Masking, provide server-side security that greatly simplify prevent unauthorized access, without the need to modify existing client applications.
- Transparent Data Encryption protects data at rest at file level, while Always Encrypted secures data in motion and at rest.

Cost

- SQL Server has a significantly lower total cost of ownership (TCO) than similar enterprise data solutions.[5] A key factor in reducing TCO is the inclusion of features with SQL Server which comparable database solutions either provide at additional cost or simply do not offer. With all the availability, performance, and security features mentioned above, SQL Server solutions can be a fraction of the cost of the competition.

Multi-OS support

- With all these benefits and capabilities available on SQL Server 2017, organizations can now choose their deployment environment.
- SQL Server 2017 is now available on more operating systems- Red Hat® Enterprise Linux®, SUSE Linux Enterprise Server, Ubuntu, and cloud and container platforms like Docker. HPE offers Linux solutions with either Red Hat Enterprise Linux or SUSE Linux Enterprise Server.

Linux native-user experience with support for exsisting SQL Server tools

Microsoft has focused on delivering a Linux-native user experience for SQL Server by providing a package-based installation, and the ability to have scripted installation with environment variables and configuration files. This is exactly what a Linux administrator wants for an easy installation and configuration experience with SQL Server.

SUSE Linux Enterprise Server

SUSE Linux Enterprise Server is a world-class, secure open source server operating system, built to power physical, virtual and cloud-based mission-critical workloads. The operating system further raises the bar in helping organizations to accelerate innovation, enhance system reliability, meet tough security requirements and adapt to new technologies.

Service Pack 3 provides full support for Direct Access (DAX) to files residing in persistent memory. Mounting file systems with the –o dax option allows directly reading from and writing to files residing on the persistent memory storage device, eliminating the overhead of the page cache to buffer reads and writes. This functionality is supported for XFS and ext4 file systems. The DAX implementation is key to obtaining optimal performance for SQL Server on Linux with HPE Scalable Persistent Memory.

HPE ProLiant DL380 Gen10 server

The HPE ProLiant DL380 Gen10 server delivers the latest in security, performance and expandability. It supports the Intel® Xeon® Processor Scalable Family with up to a 71% performance gain and 27% increase in cores[6], plus the HPE 2666 MT/s DDR4 SmartMemory supporting 3.0 TB and up to 11%[7] faster than 2400 MT/s memory. The HPE ProLiant DL380 Gen10 server has an adaptable chassis, including new HPE modular drive bay configuration options with up to 30 SFF, up to 19 LFF, or up to 20 NVMe drive options along with support for up to 3 double wide GPU options. Along with an embedded 4x1GbE, you have a choice of HPE FlexibleLOM or PCIe standup adapters which offer a choice of networking bandwidth (1GbE to 40GbE) and fabric so you can adapt and grow to changing business needs. The HPE ProLiant DL380 Gen10

[4] Always On Failover Cluster Instance requires shared storage so it cannot be utilized in a persistent memory environment.
[5] Microsoft SQL Server 2017 on Linux white paper, http://info.microsoft.com/rs/157-GQE-382/images/EN-CNTNT-SQLServeronLinux-ITDM-WhitePaper.pdf.
[6] Intel measurements. Up to 71% performance increase of Intel Xeon Platinum vs. previous generation E5 v4 average performance based on key industry-standard benchmark calculations submitted by OEMs comparing 2-socket Intel Xeon Platinum 8180 to E5-2699 v4 family processors. Any difference in system hardware or software design or configuration may affect actual performance. May 2017. Up to 27% performance increase of Intel Xeon Platinum vs. previous generation comparing 2-socket Intel Xeon Platinum 8180 (28 cores) to E5-2699 v4 (22 cores). Calculation 28 cores/22 cores= 1.27 = 27%. May 2017.
[7] The Gen10 2666 MT/s memory speed is 11% faster than that of Gen9 2400 MT/s, enabling faster server performance.

server comes with a complete set of HPE Technology Services, delivering confidence, reducing risk and helping customers realize agility and stability.

Figure 3. HPE ProLiant DL380 Gen10 server

Solution components

This paper provides configuration and performance information for HPE Scalable Persistent Memory in a Microsoft SQL Server 2017 database environment. The tests were run on an HPE ProLiant DL380 Gen10 server running SUSE Linux Enterprise Server 12 SP3. Figure 4 shows the hardware used for the persistent memory tests.

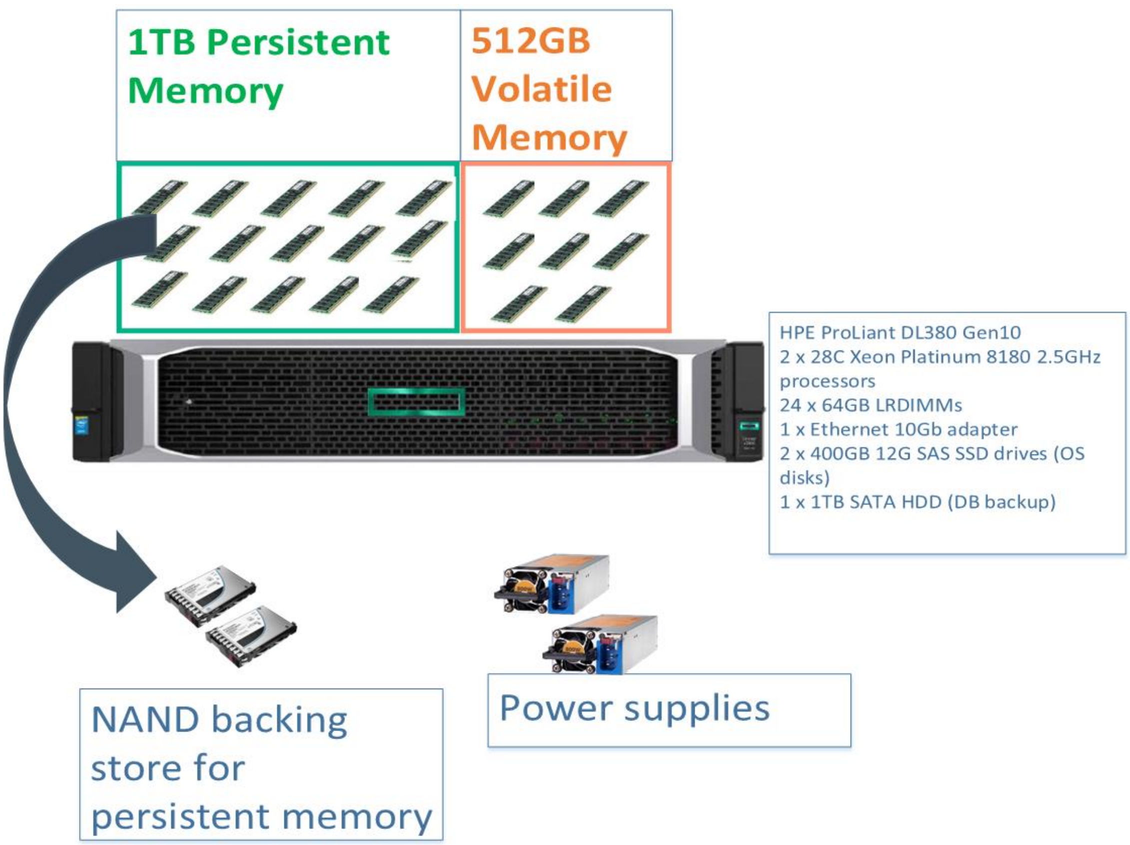

Figure 4. Hardware configuration for persistent memory tests

Hardware

Table 1 lists the details of the hardware configurations tested, including the Scalable Persistent Memory solution, as well as the platforms with traditional storage that were used for comparison testing.

Table 1. Hardware configurations used for testing solutions with HPE Scalable Persistent Memory and SSDs

	HPE ProLiant DL380 Gen10 with HPE Scalable Persistent Memory	HPE ProLiant DL380 Gen10 with SSDs	HPE ProLiant DL380 Gen9 with SSDs
Processor	Two 28-core Intel Xeon Platinum 8180 2.50GHz	Two 28-core Intel Xeon Platinum 8180 2.50GHz	Two 22-core Intel Xeon E5-2699 v4 2.20GHz
Memory	512GB volatile memory; 1TB Persistent Memory (24 x 64GB HPE DDR4 SmartMemory LRDIMMs)	512GB memory (8 x 64GB HPE DDR4 SmartMemory LRDIMMs)	512GB memory (16 x 32GB HPE SmartMemory LRDIMMs)
Ethernet	HPE Ethernet 10Gb 2-port 631FLR-SFP28 Adapter	HPE Ethernet 10Gb 2-port 631FLR-SFP28 Adapter	Embedded 1GbE Network Adapter
Local Drive	2 x 400GB 12G SAS MU SFF SSD	2 x 400GB 12G SAS MU SFF SSD	2 x 300GB 6G SAS 15K SFF HDD
Local Drive	1 x 1TB 6G SATA 7.2K rpm SFF SC Midline Hard Drive	1 x 1TB 6G SATA 7.2K rpm SFF SC Midline Hard Drive	19 x 300GB 6G SAS 15K SFF HDD
Local Drive	N/A	3 x HPE Smart Array P441/4GB 12 x 400GB 12G SAS ME SFF SSD	6 x HPE Smart Array P441/4GB 28 x 400GB 12G SAS ME SFF SSD

Table 2 shows the layout for the traditional storage and HPE Scalable Persistent Memory solutions.

Table 2. Storage layout for HPE Scalable Memory and SSD configurations

Usage	PMEM configuration	Gen10 SSD configuration	Gen9 SSD configuration	RAID configuration
O/S	2 x 400GB 12G SAS MU SFF SSD	2 x 400GB 12G SAS MU SFF SSD	2 x 300GB 6G SAS 15K SFF HDD	RAID1
Transaction logs	2 x 102GB PMEM devices (1 per socket)	2 x 400GB 12G SAS ME SFF SSD (one controller configuration) 4 x 400GB 12G SAS ME SFF SSD (three controller configuration)	8 x 400GB 12G SAS ME SFF SSD	RAID1 for PMEM RAID10 for SSDs
Data and tempDB files	2 x 282GB PMEM devices (1 per socket)	2 x 400GB 12G SAS ME SFF SSD (one controller configuration) 8 x 400GB 12G SAS ME SFF SSD (three controller configuration)	20 x 400GB 12G SAS ME SFF SSD	No RAID
Database backup	1 x 1TB 6G SATA HDD	1 x 1TB 6G SATA HDD	19 x 300GB 6G SAS 15K SFF HDD	RAID5 for Gen9 SSD No RAID for others

Best practices and configuration guidance for the solution

HPE ProLiant DL380 Gen10 server

Initial setup for the HPE ProLiant DL380 Gen10 server consisted of various BIOS and SQL settings. The following BIOS settings were modified from the default settings:

- WorkloadProfile=Custom
- CollabPowerControl=Disabled
- EnergyPerfBias=MaxPerf
- IntelDmiLinkFreq=Auto
- IntelNicDmaChannels=Enabled
- IntelPerfMonitoring=Disabled
- IntelProcVtd=Disabled
- IntelUpiFreq=Auto
- IntelUpiLinkEn=Auto
- IntelUpiPowerManagement=Disabled
- IntelligentProvisioning=Enabled
- LlcPrefetch=Enabled
- ThermalConfig=MaxCooling

SQL Server Configuration guidance

The following SQL Server trace flags were enabled for this testing:

- T652 – disable page pre-fetching scans
- T834 – use large pages
- T8744 – disable pre-fetching for Nested Loop operator

In addition, the following SQL Server settings were configured:

- disable stats collection
- min memory per query = 393216
- Max server memory = 511000MB
- Min server memory =511000MB
- Network packet size = 4096
- Lightweight pooling = 0
- Max worker threads = 0
- Max degree of parallelism = 56
- Cost threshold for parallelism = 0
- Query wait (s) = 2147483647
- Recovery interval (min) = 32767
- Default trace enabled = 0
- SQL PAL was configured through the BASH environment variable $PAL_MEMORY_SIZE=513000

- Process affinity was set to both processors
- The number of tempdb data files matched the number of logical processors, and were spread evenly across the PMEM devices or SSDs.

SQL Server Resource Governor

Note
The following settings were used for the HPE ProLiant DL380 Gen10 tests. For the HPE ProLiant DL380 Gen9, max_memory_grant_percent was set to 45%, and the default resource pools and workload groups were used.

SQL Server Resource Governor was used to configure two resource pools, two workload groups, and one classifier function to redirect incoming sessions. The first resource pool had an affinity for the first thread from each physical core. The second resource pool had affinity for the second thread of each physical core. In other words, no two sibling threads of the same physical core were part of the same resource pool.

One resource pool (pRO) was dedicated to be consumed for read-only queries. The second resource pool (pRF) was dedicated for RF1/RF2 workloads. Resource pool pRO can be memory hungry and can consume all available SQL Server memory while starving pRF. To guarantee a minimum working memory for pRF, pRO was configured to limit its memory consumption to 85% and pRF to the remaining 15% of memory. This was achieved by setting the “MAX_MEMORY_PERCENT” attribute of the respective resource pools.

Two workload groups were defined. One (gRO) was associated with pRO, and the other (gRF) with pRF. To limit the amount of memory consumed by a single query, gRO’s request_max_memory_grant_percent was set to 45%. For gRF, no capping was specified.

The benchmark client was configured to use two distinct SQL Server users. One user connection executed only the queries while the other only ran RF1/RF2 sequences. Using a classifier function, based on the SQL user, incoming sessions are pushed into either of the two workload groups.

SLES 12 SP3 kernel parameters

The following kernel parameters were modified for this testing. Note that transparent huge pages was also enabled via the SQL Server trace flag T384. The max_map_count parameter needs to be increased above the default value of 65536 on servers with large memory configurations. The last three parameters were set to reduce process migrations across NUMA nodes.

- vm.max_map_count=131072
- kernel.nmi_watchdog=0
- kernel.numa_balancing=0
- kernel.sched_nr_migrate=16
- kernel.sched_migration_cost_ns=1500000

Capacity and sizing

Workload description

Two separate types of tests were conducted, one with the TPC-H Benchmark, and the second with individual SQL queries.

TPC-H Benchmark

The TPC™ Benchmark H (TPC-H) decision support benchmark was utilized for this effort. It consists of a suite of business oriented ad-hoc queries and concurrent data modifications. The queries and the data populating the database have been chosen to have broad industry-wide relevance. This benchmark illustrates decision support systems that examine large volumes of data, execute queries with a high degree of complexity, and give answers to critical business questions. The performance metric reported by TPC-H is called the TPC-H Composite Query-per-Hour Performance Metric (QphH@Size), and reflects multiple aspects of the capability of the system to process queries. These aspects include the selected database size against which the queries are executed, the query processing power when queries are submitted by a single stream, and the query throughput when queries are submitted by multiple concurrent users. The TPC-H Price/Performance metric is expressed as $/QphH@Size. A 1000GB database was configured for this testing.

The performance metric reported by TPC-H is called the TPC-H Composite Query-per-Hour Performance Metric (QphH@Size), and reflects multiple aspects of the capability of the system to process queries. These aspects include the selected database size against which the queries are executed, the query processing power when queries are submitted by a single stream and the query throughput when queries are submitted by multiple concurrent users. The TPC-H Price/Performance metric is expressed as $/QphH@Size. To be compliant with the TPC-H standard, all references to TPC-H results for a given configuration must include all required reporting components.[8]

Individual query testing

Separate from the official, audited TPC-H benchmark results, some individual query tests were run to demonstrate the performance benefits of HPE Scalable Persistent Memory as compared to SSDs on an otherwise similarly configured HPE ProLiant DL380 Gen10 server. Each individual query was run from SQL Server Management Studio (SSMS). Prior to each query test, the impact of caching was eliminated by dropping clean buffers and the query plan cache from memory. In addition, all writes in the cache were flushed to storage and pagecache, dentries, and inodes were cleared from the file system cache.

[8] Excerpts from TPC Benchmark H Standard Specification, Revision 2.17.3 http://www.tpc.org/tpc_documents_current_versions/pdf/tpc-h_v2.17.3.pdf.

Analysis and recommendations

TPC-H Benchmark

For the TPC-H Benchmark, the results for the HPE ProLiant DL380 Gen10 with HPE Scalable Persistent Memory were compared to results from an HPE ProLiant DL380 Gen9 with traditional SSD storage. Figure 5 shows the throughput achieved on each server for three of the components of the TPC-H benchmark: QphH, Power, and Throughput. The HPE ProLiant DL380 Gen10 with Scalable Persistent Memory processed 41% more queries per hour (QphH@1000GB) than an HPE ProLiant DL380 Gen9 with SSD storage. When queries were submitted single stream (TPC-H Power @1000GB), 34% more queries per hour were processed by the HPE ProLiant DL380 Gen10 than the HPE ProLiant DL380 Gen9 server. For the tests where the queries are submitted by multiple concurrent users (TPC-H Throughput @1000GB), the HPE ProLiant DL380 Gen10 with Scalable Persistent Memory provided 48% better throughput than the Gen9 server with SSDs. This demonstrates that customers moving from older hardware can expect significantly improved throughput when deploying SQL Server with HPE Scalable Persistent Memory on Gen10 servers.

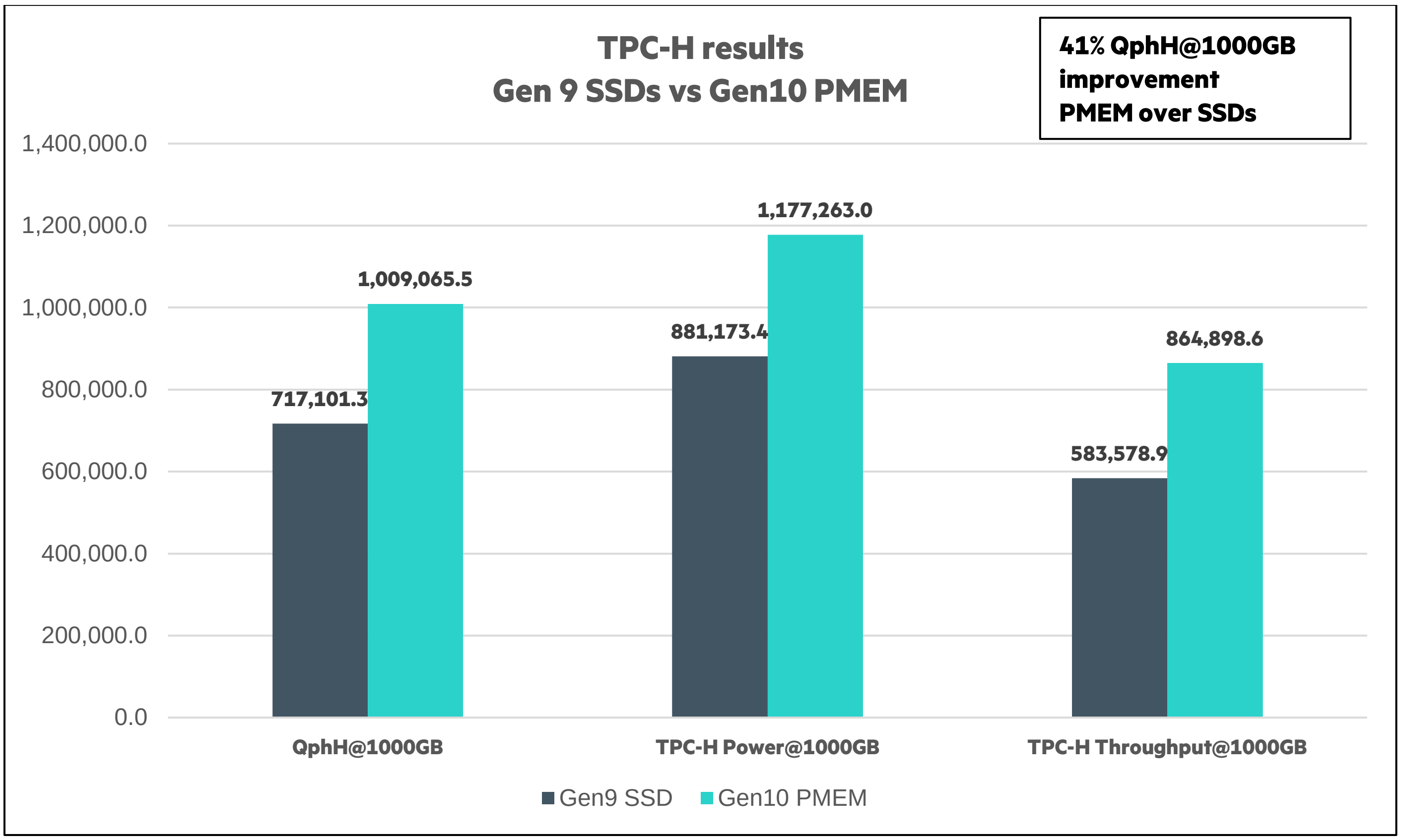

Figure 5. TPC-H results for Scalable Persistent Memory versus SSDs

The HPE ProLiant DL380 Gen10 with Scalable Persistent Memory also offers a significant reduction in hardware costs as compared to the HPE ProLiant DL380 Gen9 with traditional SSD storage. Figure 6 shows the cost of each hardware configuration, with the HPE ProLiant DL380 Gen10 with Scalable Persistent Memory configuration coming in at 31% less than the HPE ProLiant DL380 Gen9 SSD offering.

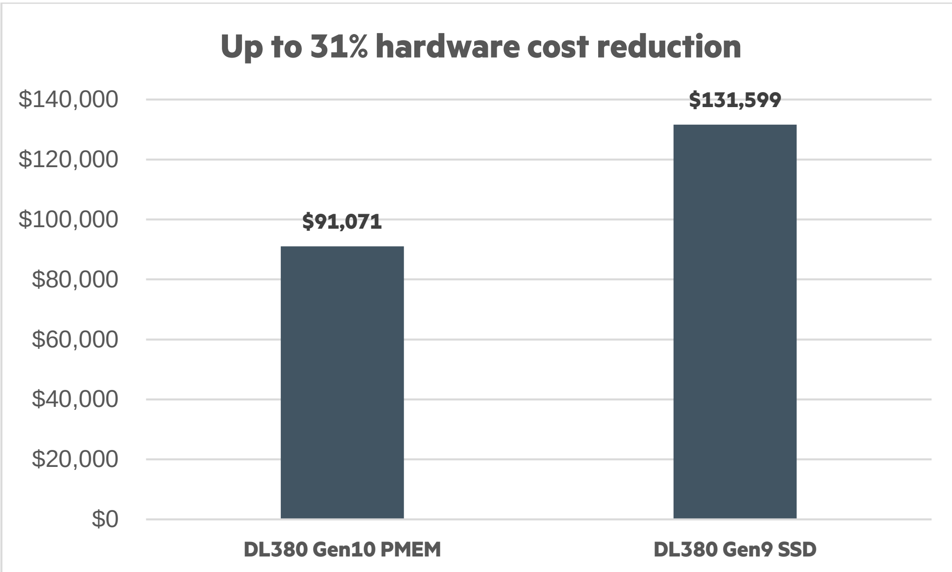

Figure 6. Hardware cost for HPE ProLiant DL380 Gen10 with persistent memory compared to HPE ProLiant DL380 Gen9 with SSD storage

In addition to the performance improvements and cost savings, this solution provides an 85% reduction in the rack space required, due to the elimination of storage. Figure 7 shows the HPE ProLiant DL380 Gen10 solution, requiring only 2U of rack space for the server alone, as compared to the HPE ProLiant DL380 Gen9 implementation with traditional storage, which had six 2U storage enclosures in addition to the server, for a total of 14U of rack space. This reduction provides for either reduced power and cooling requirements, or additional capacity for more servers.

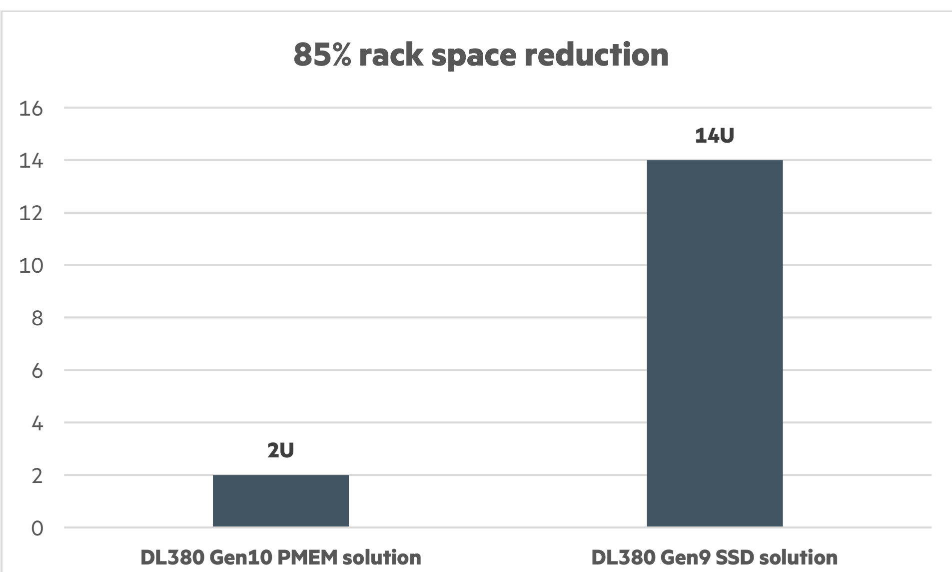

Figure 7. Rack space requirements for HPE ProLiant DL380 Gen10 PMEM and HPE ProLiant DL380 Gen9 SSD solutions

Individual query testing

Note
The individual query workloads are derived from the TPC-H Benchmark, and as such are not comparable to published TPC-H Benchmarks.

Individual query tests derived from the TPC-H Benchmark were run on an HPE ProLiant DL380 Gen10 server to demonstrate the benefits of HPE Scalable Persistent Memory when the data is not already cached. Before each test, the caches were dropped to demonstrate performance when the data is coming from storage (in this case, storage is either SSDs or HPE Scalable Persistent Memory). For these tests, the queries were run on an HPE ProLiant DL380 Gen10 server with both Scalable Persistent Memory and SSD drives. Two different SSD configurations were tested, one with a single Smart Array controller, and one with three Smart Array controllers, to demonstrate the differences in performance between various storage configurations.

The six queries highlighted here, with query IDs A through F, were derived from TPC-H Queries 1, 9, 16, 17, 18 and 19 respectively. The queries were run against a TPC-H like Wholesale Supplier schema, and are described in Table 3. As expected, the queries with the highest I/O throughput requirements were the ones that showed the most improvement when the database was residing in persistent memory.

Table 3. Descriptions of individual queries

Query ID	Query type	Description
A	I/O intensive	This query is run against a wholesale supplier schema to determine total quantity of products, their extended price, and discounts across all orders. It also computes average quantity, extended price, and discount per order. This line item table has nearly 6 billion rows and the query does a columnstore index scan, making it IO intensive. Execution plan includes a hash match to filter rows making it mildly CPU intensive.
B	I/O intensive	This query calculates overall profit made by the wholesale supplier from each country grouped by calendar year for a given product. Similar to query A, this involves columnstore index scans of multiple tables involving parts (items), their suppliers and orders (along with its line items) making it a complete I/O intensive query. Spends negligible amount of time on hash matches, joins and sorts.
C	I/O intensive and moderately CPU intensive	.A query to determine the quantity of products available from suppliers that match given criteria like a product's brand, its type, available sizes and lastly without any known customer complaints. Results are primarily grouped by product brands and ordered by count of suppliers. This query spends most of the time doing columnstore index scans on part supplier table and parts table, making it I/O intensive. Involves multiple join and aggregate functions that consume significant amount of CPU.
D	I/O intensive	This query calculates the average yearly sales amount of a product considering orders that have their order quantity value less than 20% of average quantity count across all orders of the same product. Again, similar to the query A, this query also does a columnstore index scan on the line item table and is heavily I/O intensive. A moderate amount of time is spent on sort functions that consume CPU.
E	I/O and CPU intensive	Query that lists top 100 customer orders and their total cost whose item count is above a given threshold. This is a unique query which is both I/O and CPU intensive. While it does multiple columnstore index scans of the line item table, it also consumes a lot of CPU cycles running aggregate functions on qualified orders.
F	I/O intensive	This query finds revenue generated by certain brand products that have specific criteria which are based on brand name, container type, quantity, product sizes, shipping modes and delivery types. This query does columnstore index scans of line item table and part table.

Figure 8 shows the relative performance of each query normalized to the time to complete the query on persistent memory. The query that benefitted the most from persistent memory was query F. When compared to the SSD configuration with a single controller, query F ran over 22 times faster in persistent memory. Compared to the three controller configuration, it ran 11 times faster.

Query D, which is also I/O intensive, ran almost 15 times faster in persistent memory as compared to a single controller SSD solution, and over seven times faster than the three controller SSD configuration.

The rest of the queries ran from two to seven times faster on persistent memory than on SSDs with a single controller. Compared to the three controller SSD configuration, the queries ran from 1.5 to 3.8 times faster on persistent memory.

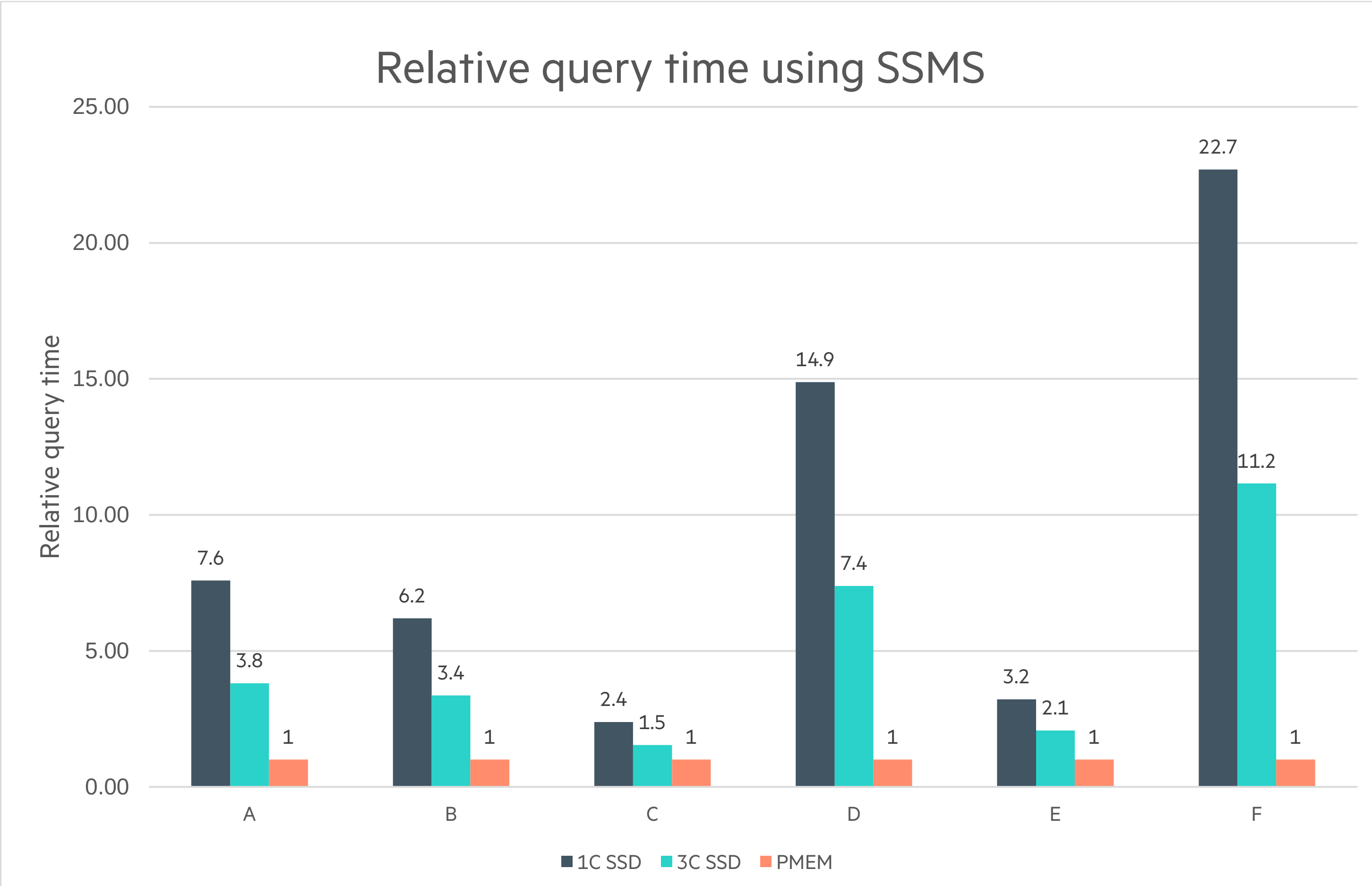

Figure 8. Relative time for selected queries, HPE Scalable Persistent Memory versus one and three controller SSD configurations

Summary

The winning combination of Microsoft SQL Server 2017 on SUSE Linux Enterprise Server with HPE Scalable Persistent Memory on HPE ProLiant DL380 Gen10 provides world-leading TPC-H Benchmark results of over 1 million QphH@1000GB, with excellent price performance of $0.47 USD per QphH@1000GB. Customers who are moving from older hardware can expect substantial cost reduction and performance improvements, including much faster query completion for I/O intensive queries. Environments where Linux is required can now deploy SQL Server, and have a lower-cost alternative to Oracle Database.

This Reference Architecture describes solution testing performed in November and December 2017.

Implementing a proof-of-concept

As a matter of best practice for all deployments, HPE recommends implementing a proof-of-concept using a test environment that matches as closely as possible the planned production environment. In this way, appropriate performance and scalability characterizations can be obtained. For help with a proof-of-concept, contact an HPE Services representative (hpe.com/us/en/services/consulting.html) or your HPE partner.

Appendix A: Bill of materials

Note

Part numbers are at time of publication/testing and subject to change. The bill of materials does not include complete support options or other rack and power requirements. If you have questions regarding ordering, please consult with your HPE Reseller or HPE Sales Representative for more details. hpe.com/us/en/services/consulting.html. Also note that, TPC rules require configuring 10% spares for disks, and including the cost of the spares in the pricing of the solution. A minimum of two spares must be configured.

Table 4. Bill of materials for HPE ProLiant DL380 Gen10 with HPE Scalable Persistent Memory

Qty	Part number	Description
		Server hardware
1	868703-B21	HPE DL380 Gen10 8SFF CTO Server (Customer to Order)
1	871619-B21	HPE DL380 Gen10 Intel Xeon Platinum 8180 (2.50GHz/28-core) Kit
1	871619-L21	HPE DL380 Gen10 Intel Xeon Platinum 8180 (2.50GHz/28-core) FIO Kit
1	876403-B21	HPE Scalable Persistent Memory 1024GB 2 Socket Server SKU (includes 1024GB persistent memory, 512GB volatile memory, battery-backed power supplies, high performance fan, 4p NVMe riser, NVMe SSDs and BIOS configuration settings)
1	817709-B21	HPE Eth 10/25Gb 2p 631FLR-SFP28 Adptr
1	GV537A8	HP 17" FlatPanel Monitor
1	RC464AA	HP PS/2 Keyboard And Mouse Bundle
1	H6J66A	HPE Universal Rack 11642 1075mm Shock Rack
1	252663-D74	HPE 24A High Voltage Core Only Corded PDU
1	H8QP7E	HPE 3Y FC 24x7 DL380 Gen10 SVC
		Storage
2	872374-B21	HPE 400GB SAS 12G MU SFF SC DS SSD
2	872374-B21	HPE 400GB SAS 12G MU SFF SC DS SSD (10% spares)
1	655710-B21	HPE 1TB 6G SATA 7.2K rpm SFF (2.5-inch) SC Midline Hard Drive
2	655710-B21	HPE 1TB 6G SATA 7.2K rpm SFF (2.5-inch) SC Midline Hard Drive (10% spares)

Resources and additional links

HPE Data and Analytics Reference Architectures, hpe.com/info/dm-ra

HPE ProLiant DL380 Gen10 server, hpe.com/servers/dl380

HPE Persistent Memory, hpe.com/info/persistentmemory

Microsoft SQL Server on Linux, https://docs.microsoft.com/en-us/sql/linux/sql-server-linux-overview

Microsoft SQL Server 2017 on Linux, http://info.microsoft.com/rs/157-GQE-382/images/EN-CNTNT-SQLServeronLinux-ITDM-WhitePaper.pdf

SQL Server on Linux Frequently Asked Questions (FAQ), https://docs.microsoft.com/en-us/sql/linux/sql-server-linux-faq

Microsoft SQL Server 2017 on Linux Quick Start Guide,
http://info.microsoft.com/rs/157-GQE-382/images/EN-US-CNTNT-Whitepaper-DBMC-SQLServerOnLinuxQuickStart_Updated.pdf

Microsoft SQL Server 2017, https://www.microsoft.com/en-us/sql-server/sql-server-2017

Diskless (yes diskless!) database with HPE scalable persistent memory Blog,
https://community.hpe.com/t5/Servers-The-Right-Compute/Diskless-yes-diskless-database-with-HPE-scalable-persistent/ba-p/6989039#.WnETtljwZaQ

SUSE Linux Enterprise Server 12 SP3 Release Notes, https://www.suse.com/releasenotes/x86_64/SUSE-SLES/12-SP3/

To help us improve our documents, please provide feedback at hpe.com/contact/feedback.

Sign up for updates

Hewlett Packard Enterprise

a00041643enw, February 2018