IN&OUT AG

Oracle Performance Benchmark HPE Intel RHEL and IBM Power8 AIX

Andreas Zallmann
CEO, In&Out AG

Version:	1.00
Date:	18/11/2016
Classification:	Not classified

In&Out AG IT Consulting & Engineering

Seestrasse 353, CH-8038 Zurich
Phone +41 44 485 60 60
Fax +41 44 485 60 68

info@inout.ch, www.inout.ch

Foreword

The present study was co-funded by HPE, Intel and Red Hat. Half of the service costs involved were covered by these companies. In&Out provided the remaining half of the services free of charge and made available the licences for OraBench™ free of charge.

As an independent consultancy firm, In&Out ensures that the co-funding companies have not exerted any influence on the measurements and the contents of this study and that these were produced independently. There is no financial relationship between In&Out and the aforementioned companies.

Background

At present, IBM POWER systems with the AIX operating system are very commonly used for central banking or insurance solutions based on Oracle databases in medium-sized and large businesses. In the field of application servers and web servers, x86-based systems on Linux (generally Red Hat Enterprise Linux, RHEL) have come into widespread use in data processing centres. Many customers are in the process of deciding on the platform on which they want to operate their central database systems and core applications in the future.

In addition to aspects such as costs, stability, scalability, operation and others, the performance aspect also always plays an important role. Furthermore, publicly available CPU benchmarks such as SPECint are only meaningful to a limited degree, as they only measure pure CPU performance; however, the performance of the entire system is the determining factor. Benchmarks that first and foremost aim to set new performance records, such as TPC-C or SAPS, are likewise only suitable to a certain extent. These benchmarks are reached with non-representative, maximum use of hardware. In addition, it is often the case that current systems are not represented in the benchmark.

With the OraBench™ performance benchmark suite, In&Out possesses a representative and powerful Oracle benchmark that can be used to compare the performance of Oracle database systems with all relevant indicators on different platforms. In&Out has measured Oracle performance with OraBench™ on current x86 servers with a range of virtualisation technologies and on current IBM Power8 servers. The company HPE made the relevant hardware and environments available to us in the European benchmark centre in Böblingen (Germany). The Power systems were measured in a customer environment.

This document serves as the basis for architecture- and manufacturer-related decisions concerning Oracle platforms, such as Avaloq, Finnova, Temenos T24, Adcubum Syrius or other Oracle-based applications.

In&Out AG

The consultancy and engineering firm In&Out (www.inout.ch) based in Zurich, provides consultancy services in the key areas of technology and platforms, cloud computing and data centre & continuity management; these services are independent of manufacturers and impartial. In&Out does not sell any products from third-party manufacturers; however, it does foster technology partnerships with a wide range of leading manufacturers, including; HPE, IBM, Red Hat, EMC, HDS, VMware and others.

In&Out often advises its customers in platform-related decisions and focuses mainly on the aspect of performance benchmarks. OraBench™, the Oracle benchmark suite, was developed over 10 years ago by In&Out and has been used by well-known customers for more than 50 performance benchmarks.

About the author

Andreas Zallmann,
andreas.zallmann@inout.ch
In&Out AG, Seestrasse 353, 8038 Zurich
www.inout.ch

Andreas Zallmann studied Information Technology at the University of Karlsruhe and has been employed at In&Out AG since 2000. He is responsible for the business area of technology and was appointed CEO of In&Out AG on 1 October 2016. He is the developer of the In&Out performance benchmarking tools IOgen™ (storage I/O benchmarks), NETgen™ (network benchmarks) and CPUgen™ (CPU Benchmarks), and he has carried out benchmarking for numerous customers and manufacturers over the last few years.

In&Out possesses many years of practical experience in the architecture, design, benchmarking and tuning of storage and system platforms, particularly for core applications for banks and insurance companies.

Management Summary

The CPU performance of contemporary HPE servers with current Intel processors using RHEL has now surpassed the contemporary Power 8 systems using AIX, and this is especially apparent in the Oracle PL/SQL benchmarks.

The tests with very high I/O throughput are not genuinely comparable between IBM and HPE Intel RHEL, because the I/O connection of these servers was considerably higher in the test than that of the IBM systems. However, it is evident that very high data writes and data scan rates can be achieved on all HPE Intel RHEL systems, which, with 5 GB/s, would be sufficient for very high requirements.

The HPE Intel RHEL stack also demonstrates very good performance in other tests, which exceeded the performance of the IBM platform in practically all tests. In a number of tests, however, a significant performance impact was evident as a result of virtualisation, and this was especially high for a relatively large number of parallel small I/Os. Generally speaking, VMware ESX has a lower impact in this respect than the Red Hat virtualisation RHV.

In conclusion, the use of Oracle databases on x86 systems with Red Hat Linux is recommended from a performance perspective. This is particularly true in view of the fact that performance is higher per core, but only half the licence fees are incurred per Intel core (licence factor 0.5) compared with Power cores (licence factor 1.0).

We must remember that IBM Power platforms offer excellent virtualisation, and cope very well even with very high CPU loads up to 80% and with a very large number of virtual systems. The x86 systems with VMware or RHV usually work with lower usage. However, the hardware costs for Intel systems are several times lower than those of the Power systems.

The aim of this benchmark is to perform a comparison with the same use of resources based on 8 physical cores. This involved measuring 2 different Intel CPUs: the E5 family (maximum of 2 sockets) and the E7 family (2 to 16 sockets).

Benchmark setup

System platforms

The following platforms were included in the comparison:

- IBM Power8 mid-range systems Power E870 on IBM AIX
- HPE DL380 Gen9 on Red Hat Enterprise Linux (RHEL)
- HPE DL580 Gen9 on Red Hat Enterprise Linux (RHEL)
- The HPE servers were each measured as bare metal (BM) without virtualisation and with VMware ESXi and Red Hat virtualisation (RHV).

A performance benchmark with the HPE Superdome X (SDX) was deliberately not performed, because this uses the same processors as the HPE DL580 Gen9 (E7 family), though scaled up to 16 sockets, and the aim of this test was to compare systems with just 8 cores (equal to 2 CPU sockets with Intel).

The exact test configuration can be found in the table below:

	IBM Power 8	**HPE DL580 Gen9**	**HPE DL380 Gen9**
System	IBM E870 MME	Proliant DL580 Gen9	Proliant DL380 Gen9
CPU	8 IBM Power8 4.2 GHz up to 80 cores	4 Intel Xeon E7-8893 v4 3.20 GHz, boost 3.50 GHz 16 cores	2 Intel Xeon E5-2637 v4 3.50 GHz, boost 3.70 GHz 8 cores
Maximum configuration	8 sockets, 80 cores	4 sockets, 16 cores	2 sockets, 8 cores
Tested configuration	8 cores	2 sockets, 8 cores	2 sockets, 8 cores
Virtual processors	8 VP, 4 EC	8 VP	8 VP
Threads	SMT4, 32 threads	Hyperthreading 16 threads	Hyperthreading 16 threads
Memory	128 GB	128 GB	128 GB
OS	AIX 7.1	RHEL 7.2	RHEL 7.2
Storage	HDS G1000 SSD tier 8 LUNs	HPE 3PAR StorServ 8450 All flash 16 LUNs	HPE 3PAR StorServ 8450 All flash 16 LUNs
Storage connection	4 x 8 Gbit	8 x 16 Gbit	8 x 16 Gbit
Virtualisation	PowerVM	Bare metal VMware ESX6 U2 RHV 4.0	Bare metal VMware ESX6 U2 RHV 4.0
Oracle	12.1.0.2.0	12.1.0.2.0	12.1.0.2.0

Table 1 – Benchmark setup

The HPE DL580 Gen9 and DL380 Gen9 systems will be referred to as DL580 and DL380 below.

All measurements were performed with virtual or physical systems, each with 8 physical cores and active multithreading (IBM 4-way simultaneous multithreading SMT-4, Intel Xeon 2-way hyperthreading). On the IBM servers, the cores were defined with all threads as virtual processors (i.e. 8 VP = 32 threads with SMT-4); on the x86 servers, the threads were each configured for ESX and RHV (for 2-way hyperthreading, i.e. 16 threads and 8 cores). The DL380 features exactly 8 cores; in the case of the DL580 with 16 cores, VM pinning was used to ensure that the VM only runs on 8 cores with 16 threads and does not use the other remaining 8 cores. In the bare metal tests on DL580, 2 CPUs were physically removed for these tests, so that only 8 cores were used.

On IBM systems of the Power8 class, 2-way or 8-way multithreading is still available. Comparison tests with SMT-8 did not reveal any performance improvements, and showed performance deterioration in some areas; for this reason, the test was performed with SMT-4, as this showed the best throughput.

This benchmark therefore shows performance with 8 physical cores, irrespective of the maximum possible configuration. It must be pointed out that one Oracle Enterprise Edition licence is required for each IBM P8 core, while only 0.5 Oracle licences are charged per Intel x86 core. This means that, for the same number of cores, only half as many Oracle licences are required on x86 servers as on Power systems.

According to SPECint, the performance of the IBM Power8 and Intel Xeon E5-2637 v4 (used in DL380) processors is the same with 60 SPECint_base_rate per core. The Intel CPU E7-8893 v4 is slightly below that with 53.5 SPECint_base_rate (source: www.spec.org).

In the benchmarks compared here, the storage connection has a significantly more powerful configuration for the HPE systems than for the IBM systems. In the IBM set-up, the maximum theoretical throughput with 4 x 8 Gbits/s is 4 GB/s, while in the HPE set-up it is theoretically 16 GB/s with 8 x 16 Gbps. We will refer to this in this document where these different storage connections play a role.

Benchmarking tool

The benchmarks were performed using the In&Out OraBench™ 6.9 benchmarking tool. OraBench™ is able to measure different Oracle key performance indicators fully automatically and determines the transaction performance, I/O figures, CPU and memory load, as well as various other important indicators on the benchmarking server.

Benchmarking profile

The benchmark was performed using the OraBench database size L (LARGE). The key parameters can be found in the table below:

Parameter	N	S	M	L	VL	XL
Description	Notebook	Small	Medium	Large	Very Large	Extra Large
# cores	1-4	4-8	8-16	16-32	32-64	> 64
RAM	4-8 GB	8-64 GB	32-128 GB	64-256 GB	128-512 GB	> 512 GB
Database size	64 GB	128 GB	256 GB	512 GB	1 TB	2 TB
SGA total	1 GB	4 GB	16 GB	64 GB	256 GB	> 256 GB
db_cache_size	0.6 GB	3 GB	12 GB	42 GB	168 GB	
shared_pool_size	128 MB	256 MB	512 MB	1 GB	2 GB	
large_pool_size	128 MB	256 MB	512 MB	1 GB	2 GB	
Log buffer	2 MB	8 MB	16 MB	32 MB	64 MB	
pga_aggregated_target	256 MB	1 GB	4 GB	8 GB	> 8 GB	
job_queue_processes	32	64	128	256	512	1024
parallel_max_servers	64	128	256	512	1024	2048
db_writers	1	2	4	8	16	>= 16
Processes	128	256	512	1024	2048	4096

Table 2 – OraBench database sizes

The following table summarises the 8 test classes of OraBench (T100-T800). It describes for each test class which typical database operations are covered, how many tests are performed in this test class, what the key performance indicators are and for which resources the load is especially high.

Test class	Real-world database operations	# tests	Key Performance Indicators	Resource
T100	PL/SQL tests	9	CPU speed CPU throughput	CPU
T200	Data-write	1	I/O throughput	Storage
T300	Data load (sequential)	7	Load rate	CPU/storage
T400	Data scan (sequential)	8	I/O throughput scan rate	CPU/storage
T500	Data aggregation	5	Processing time	CPU/storage

Test class	Real-world database operations	# tests	Key Performance Indicators	Resource
T600	Data select (random)	12	Transaction service time Transaction throughput	CPU/storage
T700	Data update (random)	6	Transaction service time Transaction throughput I/O throughput	CPU/storage
T800	Data join	2	Transaction service time I/O throughput	CPU/storage

Table 3 – OraBench tests

In total, all 50 tests were performed on all measured systems with increasing parallelism. The test set-up in OraBench was identical for all systems. During the tests, the relevant Oracle parameters were recorded in AWR reports and the system load was ascertained. For reasons of space, especially important and representative tests were selected for the purposes of illustration in this whitepaper.

A number of tests are marked with * or **. These concern optional tests that do not necessarily have to be performed.

Explanatory notes for the graphs

Each graph shows seven curves:

- Blue: IBM P8 / AIX
- Green: HPE DL380 G9/RHEL (3 curves)
- Yellow: HPE DL580 G9/RHEL (3 curves)

Three curves are shown for each of the HPE x86 servers:

- Continuous line: RHEL on bare metal (without virtualisation)
- Dashed line: RHEL on VMware ESXi
- Dotted line: RHEL on RHV (Red Hat Enterprise Virtualisation)

The X-axis shows the parallelisms increasing from 1, while the Y-axis shows the respective measurement values. These are generally kOps/s, i.e. 1000 operations per second. A value of 2,500 kOps therefore means 2.5 million operations per second. In individual test series, the I/O throughput in MB/s is also given. In principle, a higher value is ‘better’.

Benchmarks on Oracle databases are benchmarks on a high-complexity system. The database performs certain operations at certain times, which cannot always be fully controlled, such as redo log file switches; many operations are carried out asynchronously in the background and the cache behaviour is not entirely deterministic. This may indeed give rise to certain variations in a number of tests. In order to identify these influences and overall performance, the tests are, as a rule, run with increasing parallelism and the development of the performance is assessed with increasing parallelism.

Explanatory notes for the heat maps

The heat maps are shown in 2 forms:

- The speed heat map shows the best single thread performance in parallelism 1
- The throughput heat map shows the best throughput in any parallelism

The best value in each case is always shown in dark green, and the poorer values are shown in gradients through bright green, yellow, orange and red. Red is always used for the value ‘0’. The colouring is based on a standard colour scale in Microsoft Excel.

Results of the T100 PL/SQL tests

For the T100 PL / SQL tests, we present the three most important tests from this series. The other tests show generally similar results. The PL/SQL tests are very heavily dependent on CPU performance.

T132 PL/SQL arithmetic mix number

This test carries out a mixture of arithmetic operations on the Oracle data type NUMBER. This integer data type codes a number into a half-byte (4 bit) each time and calculates using these half-bytes. The calculation method is broadly similar to a written addition. The number data type is therefore relatively slow compared with a direct arithmetic operation based on an integer or floating point unit. The NUMBER data type is commonly used, as it is implemented in the Oracle standard libraries and not in a platform-specific way. This guarantees maximum portability.

When the test is run, 38 operations are carried out: 1 random generation (2.6%), 8 additions (21%), 7 subtractions (18.5%), 4 multiplications (10.5%), 3 divisions (7.9%), 6 ABS functions (15.8%), 3 SQRT functions (7.9%), 3 POWER functions (7.9%), and 3 LN functions (7.9%). When parallelism increases, the number of processors and threads can be used efficiently because different arithmetic operations are carried out at the same time.

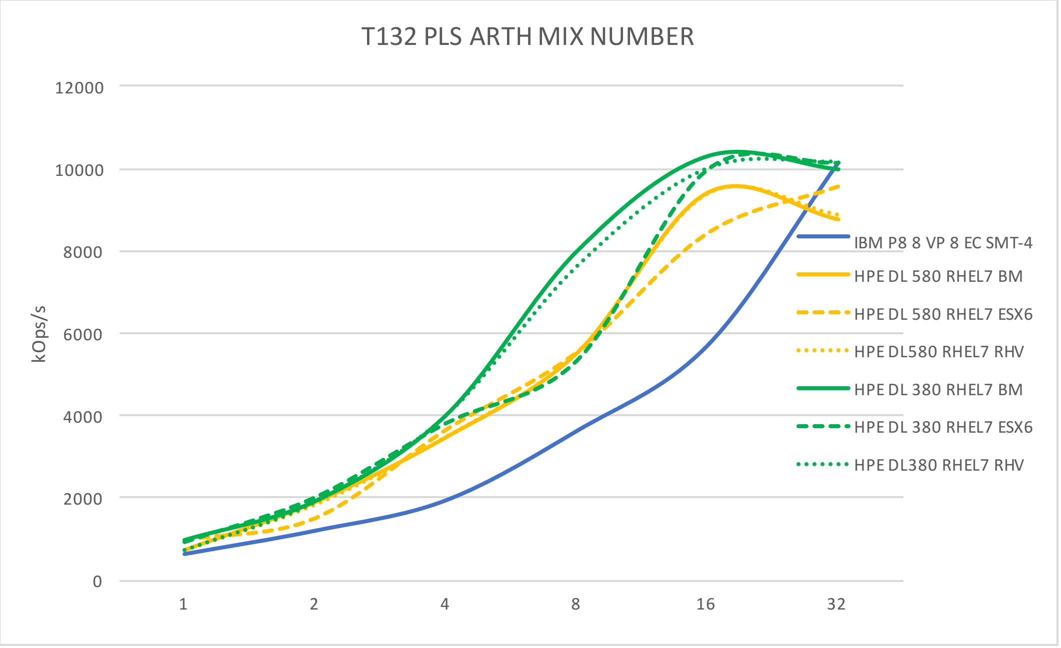

Figure 1 – T132 PL/SQL arithmetic mix number

The results of the tests show that as parallelism increases, the processing performance increases continuously up to the maximum number of available threads. For IBM Power this is 32 threads (8 cores x 4 threads), and for the x86 systems this is 16 threads (8 cores x 2 threads). In this test, the capacity of multithreading is used in full.

It is noticeable that at the same level of parallelism, the x86 systems are significantly faster than the Power systems. The single thread performance in one process is already up to 40% higher; at 16, the optimum parallelism for the x86 systems, the performance gain is around 80%. However, the IBM system continues thereafter on the scale up to parallelism 32, while the x86 systems with only 16 threads no longer achieve any increase in performance. The overall performance per core is similar for all systems, although the Power processors require double the number of processes for this.

There are no significant performance losses in the different virtualisations on x86 compared with a bare metal set-up. In principle, (as in all CPU-intensive tests), the DL380 server performs slightly better with the somewhat higher clock speeds of the E5-2637 processor.

The utilised CPU capacity at maximum parallelism is 90 to 100% in all tests. In the x86 systems, the CPU usage is already so high at 16 processes, while in the IBM systems it is only barely 50% due to the higher number of threads.

T133 PL/SQL arithmetic mix float

This test performs a mixture of arithmetic operations on the Oracle data type FLOAT. This floating point data type is generally calculated by direct arithmetic operations on the floating point units of the CPU. It is therefore markedly faster than the NUMBER data type and reaches almost 8 times the throughput (80 kOps instead of 10 kOps).

As with the NUMBER data type, 38 operations are performed while the test is run, which are divided up in the same way. When parallelism increases, the number of processors and threads can be used efficiently because different arithmetic operations are carried out at the same time.

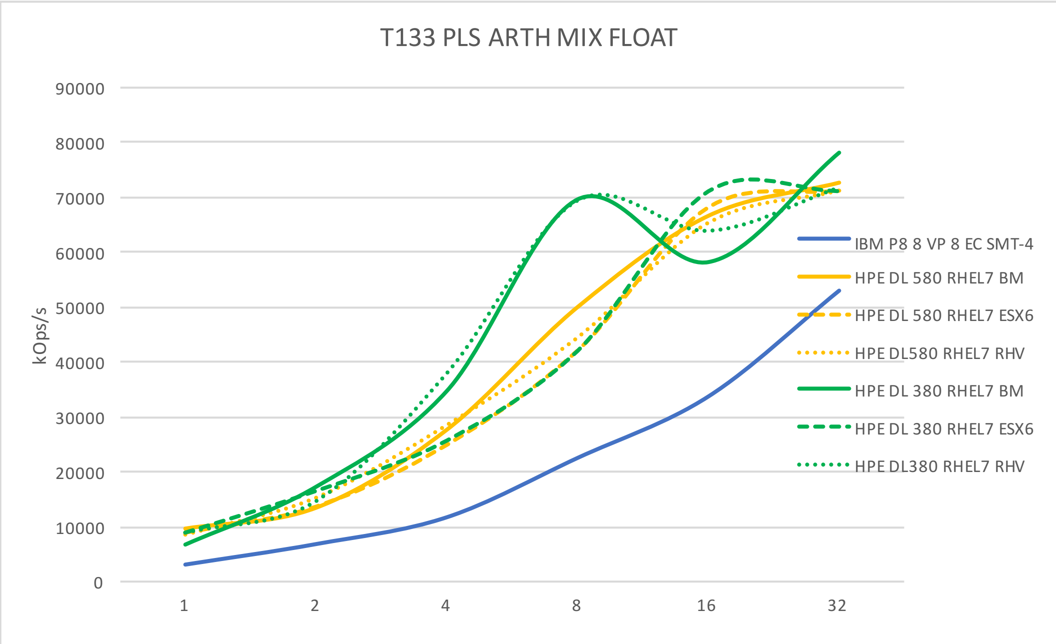

Figure 2 - T133 PL/SQL arithmetic mix float

The results of the tests show that as parallelism increases, the processing performance increases continuously up to the maximum number of available threads. For IBM Power this is 32 threads (8 cores x 4 threads), and for the x86 systems this is 16 threads (8 cores x 2 threads). In this test, the capacity of multithreading is used in full.

It is noticeable that at the same level of parallelism, the x86 systems are significantly faster than the Power systems. The single thread performance in one process is already up to 100% higher; at 16, the optimum parallelism for the x86 systems, the performance gain is also 100%. However, the IBM system continues thereafter on the scale up to parallelism 32, while the x86 systems with only 16 threads no longer achieve any increase in performance. The overall performance on the x86 systems is nevertheless a good 30% higher.

There are no significant performance losses in the different virtualisations on x86 compared with a bare metal set-up. In principle, (as in all CPU-intensive tests), the DL380 server performs slightly better with the somewhat higher clock speeds of the E5-2637 processor.

The utilised CPU capacity at maximum parallelism is 90 to 100% in all tests. In the x86 systems, the CPU usage is already so high at 16 processes, while in the IBM systems it is only barely 50% due to the higher number of threads.

T141 PL/SQL string

These tests carry out a mixture of string operations on the Oracle data type VARCHAR2 (32). When the test is run, 29 operations are carried out: 1 random string generation (3.4%), 6 INSTR (20.7%), 9 CONCAT (31.0%), 3 LENGTH (10.3%), 3 SUBSTR (10.3%), 1 CHR (3.4%), 5 UPPER (17.2%), 1 MOD (3.4%). When parallelism increases, the number of processors and threads can be used efficiently because different string operations are carried out at the same time.

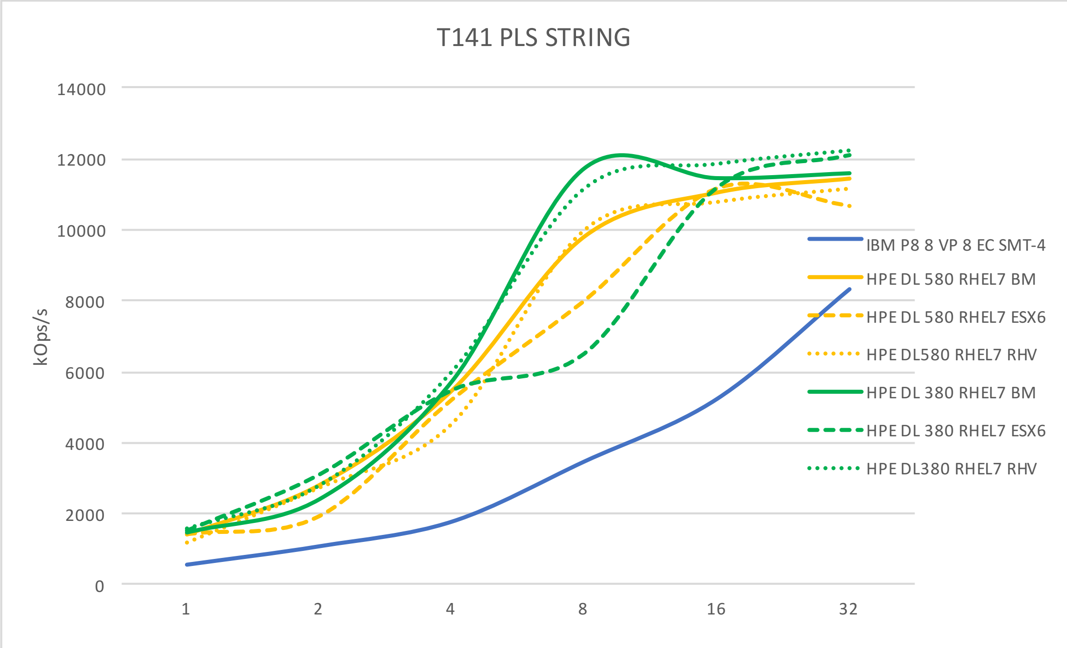

Figure 3 - T141 PL/SQL String

The results of the tests show that as parallelism increases, the processing performance increases continuously up to the maximum number of available threads. For IBM Power this is 32 threads (8 cores x 4 threads), and for the x86 systems this is 16 threads (8 cores x 2 threads). However, in the x86 systems, the maximum processing performance is generally already achieved with 8 processes; the effects of multithreading do not have much of an effect on x86 in this test.

It is noticeable that at the same level of parallelism, the x86 systems are significantly faster than the Power systems. The single thread performance in one process is already up to 100% higher; at 8, the optimum parallelism for the x86 systems, the performance gain is also 100%. However, the IBM system continues thereafter on the scale up to parallelism 32, while the x86 systems no longer achieve any significant increase in performance beyond 16 parallel processes. The overall performance on the x86 systems is nevertheless a good 35% higher.

There are no significant performance losses in the different virtualisations on x86 compared with a bare metal set-up. In principle, (as in all CPU-intensive tests), the DL380 server performs slightly better with the somewhat higher clock speeds of the E5-2637 processor.

The utilised CPU capacity at maximum parallelism is 90 to 100% in all tests. In the x86 systems, the CPU usage is already so high at 16 processes, while in the IBM systems it is only barely 50% due to the higher number of threads.

Summary of T100 PL/SQL tests

The T100 tests are very CPU-intensive tests. The effect on virtualisation is minimal in each case.

The table below shows the single thread performance with parallelism 1 (=speed) for the platforms. We can see that the DL380 with the 3.5 GHz processor is the fastest in the majority of cases, closely followed by the DL580 with the 3.2 GHz processor. In single thread performance, the performance of the IBM P8 is significantly lower despite the faster clock speed of 4.2 GHz.

Tests	Description	IBM P8	DL580 BM	DL580 VM	DL580 RV	DL380 BM	DL380 VM	DL380 R'
T111	PL SQL Loop	10204	18519	22727	20833	21739	22727	22727
T121	PL SQL Random Number	176	417	455	463	510	472	521
T122	PL SQL Random String generation	22	53	58	53	61	63	63
T131	PL SQL arethmetic mix, integer	556	952	833	952	769	1000	1000
T132*	PL SQL arethmetic mix, number	645	741	667	909	952	1000	952
T133*	PL SQL arethmetic mix, float	3175	9524	6061	7143	6897	9524	9524
T134*	PL SQL arethmetic mix, double	3846	6061	10000	6250	10000	10000	6667
T141*	PL SQL string mix	561	1395	1395	1429	1463	1579	1579
T151*	PL SQL finbonacci number n=39	32	20	20	20	22	20	24

Table 4 - T100 PL/SQL Speed heat map (higher values = better, except for T151)

The following table shows the maximum throughput for each of the 8 physical cores. It shows a similar picture to the speed test, although the IBM P8 is on a par with the HPE X86 RHEL results in a number of tests (T131-133) and is at least markedly closer to those results in the other tests. Here is where the advantage of the SMT-4 of the IBM P8 server has an effect compared with the 2-way hyperthreading of the Intel CPU. However, the Intel-based servers still demonstrate a significantly higher throughput in most tests overall.

Tests	Description	IBM P8	DL580 BM	DL580 VM	DL580 RV	DL380 BM	DL380 VM	DL380 R'
T111	PL SQL Loop	153846	172043	170213	166667	179775	181818	181818
T121	PL SQL Random Number	2649	3828	3687	3604	3922	4000	3941
T122	PL SQL Random String generation	337	451	456	438	487	478	479
T131	PL SQL arethmetic mix, integer	10000	9697	9697	9143	10323	10159	10159
T132*	PL SQL arethmetic mix, number	10159	9412	9412	8312	10323	10323	9846
T133*	PL SQL arethmetic mix, float	52893	21	33	28	29	21	21
T134*	PL SQL arethmetic mix, double	57658	76190	71111	72727	82051	80000	72727
T141*	PL SQL string mix	8348	11429	11163	10847	11707	12229	12000

Table 5 - T100 PL/SQL throughput heat map

Results of T200 data write

This test installs new database files in Oracle with a size of 8 GB each. First of all, 1 file of 8 GB is installed, followed by 2, 4, 8 and 16 files in parallel, therefore adding up to 31 files totalling 248 GB.

T211 data write

This test determines the sequential write performance of Oracle. Sequential writes occur in Oracle e.g. in the redo log writer process LGWR, RMAN backup and restore, export, data pump, redo log archiver ARCH, changing sort areas in TEMP Tablespace and the recovery writer RVWR, and are an important indicator for the efficiency of the whole I/O stack.

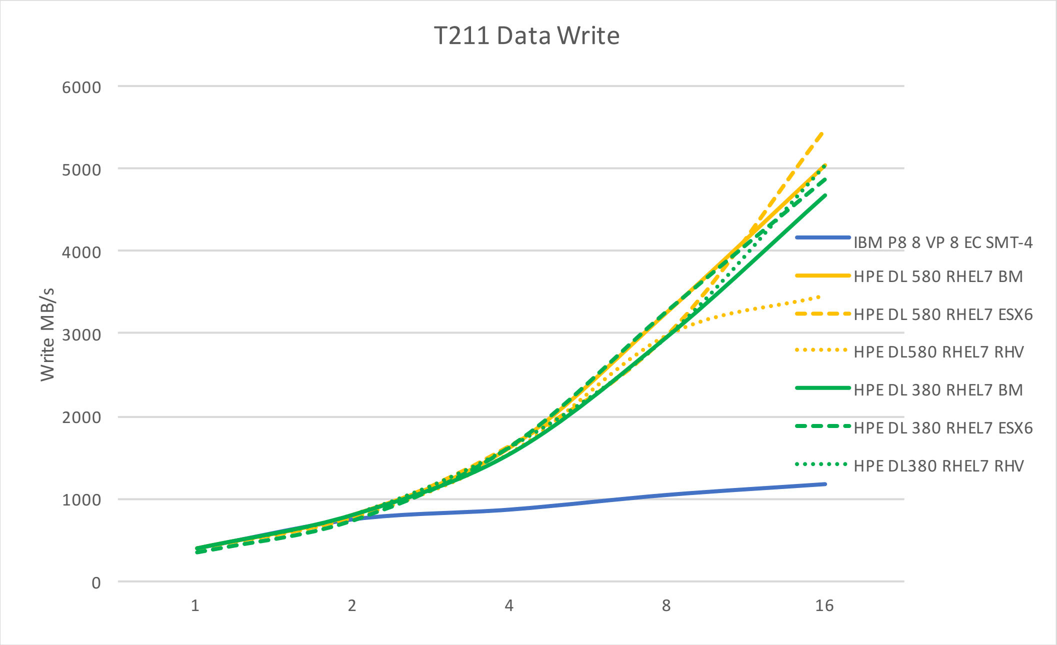

Figure 4 - T211 data write

The results of this test show that sequential write performance increases as parallelism increases. In the HPE Intel RHEL systems, this increases virtually linearly up to a bandwidth of 5 GB/s, which Oracle can effectively write to the storage, while the IBM Power system only reaches a good 1 GB/s. It must be pointed out, however, that the storage connection of the IBM system was significantly lower at 4 x 8 Gbit (maximum theoretical bandwidth of 4 GB/s) than with the HPE systems at 8 x 16 Gbit (maximum theoretical bandwidth of 16 GB/s). A direct comparison of data write performance is therefore not permissible; all systems occupy approximately 30% of the theoretical storage bandwidth.

The processing performance could certainly increase significantly with all systems if even higher parallelism were used. The CPU performance on the IBM is in the range of below 5%, while for the HPE systems it is 5-13% with 4 times the write performance.

The essential insight from this test is that all HPE Intel RHEL platforms can write data in Oracle at 4-5 GB/s and can also do this with active virtualisation. These are impressive figures that are rarely necessary in reality.

Results of T300 data load

The T300 tests simulate the insertion of large quantities of data to an Oracle database. This is done by copying a table based on 2 basic types:

- T310 and T320: Conventional load:
  Commit after every 2 rows, last test after every 10 rows, typical for OLTP environments
- T330: Bulk load: commit at the end of each copy process, typical for DWH load

A further distinction is also made between the type of table:

- T310: Small table, non-partitioned
- T320: Small table, list-partitioned
- T330: Large table, range-list partitioned

In addition, different storage structures are also tested:

- Heap-organised, test 311, 321, 331 (uncompressed) and 332 (compressed)
- Index-organised, test 312, 322
- Hash-cluster organised, test 313

T321 Data load conventional heap-organised, list-partitioned

The test performs a conventional insert in which a commit takes place after every 2 rows. This is carried out using a parallelism of 1-8. The test with parallelism 1 is performed twice so that the number of inserted rows tallies. The last test with parallelism 8 is repeated with a Commit after every 10 rows. This multiplies the throughput.

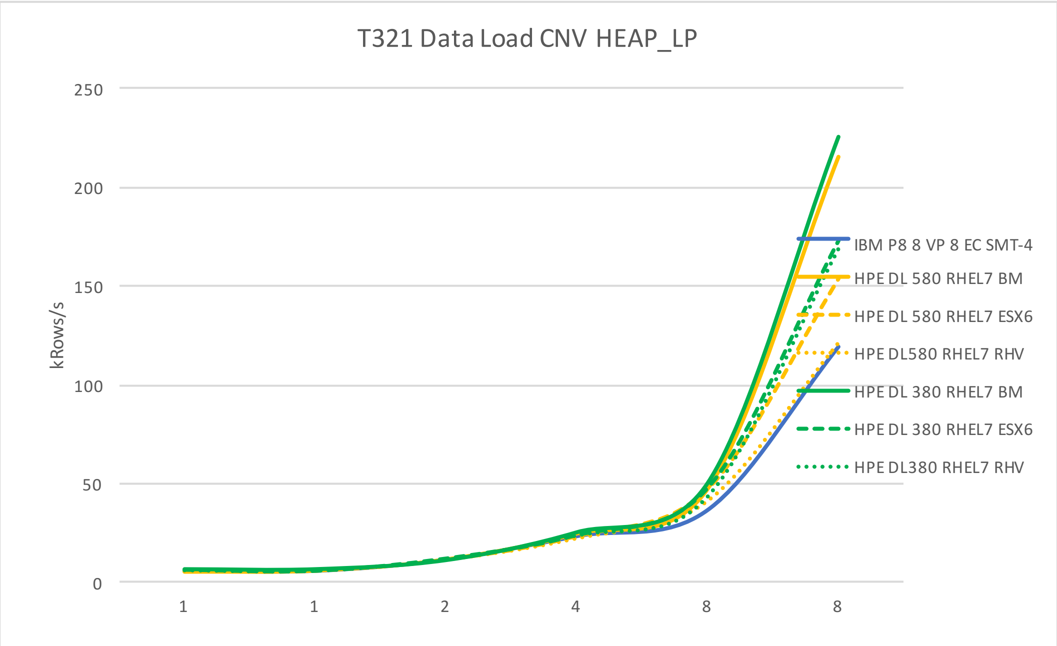

Figure 5 - T321 data load conventional heap-organised, list-partitioned table

Initially with 2 rows per Commit, there are no significant differences apparent between any of the platforms. In this stage, the limitation of processing the extremely high number of Commits lies internally within the Oracle database. It is only when it changes to 10rows per Commit that the differences have an effect. The fastest systems in this test are the 2 HPE bare metal systems without virtualisation. Virtualisation in VMware ESXi and RHV achieves somewhat lower maximum values. The IBM values are at the lower end of this range.

None of the systems are able to utilise the maximum number of threads in full; this is significantly more noticeable with the IBM platform with 32 threads compare to the HPE systems with 16 threads. Therefore, the CPU usage on the HPE systems is 50%, while it is only 25% on the IBM systems. It is to be expected that the IBM system would catch up with the throughput if the parallelism were higher. The I/O connection does not play any important role in this test, as the maximum bandwidths of 500 MB/s are too small.

T331 data load bulk heap-organised, range-list-partitioned

A bulk insert with a parallelism of 1-16 is carried out with a Commit at the end each time. The test with parallelism 1 is performed twice at the start, so that the total number of inserted rows corresponds to the requirements.

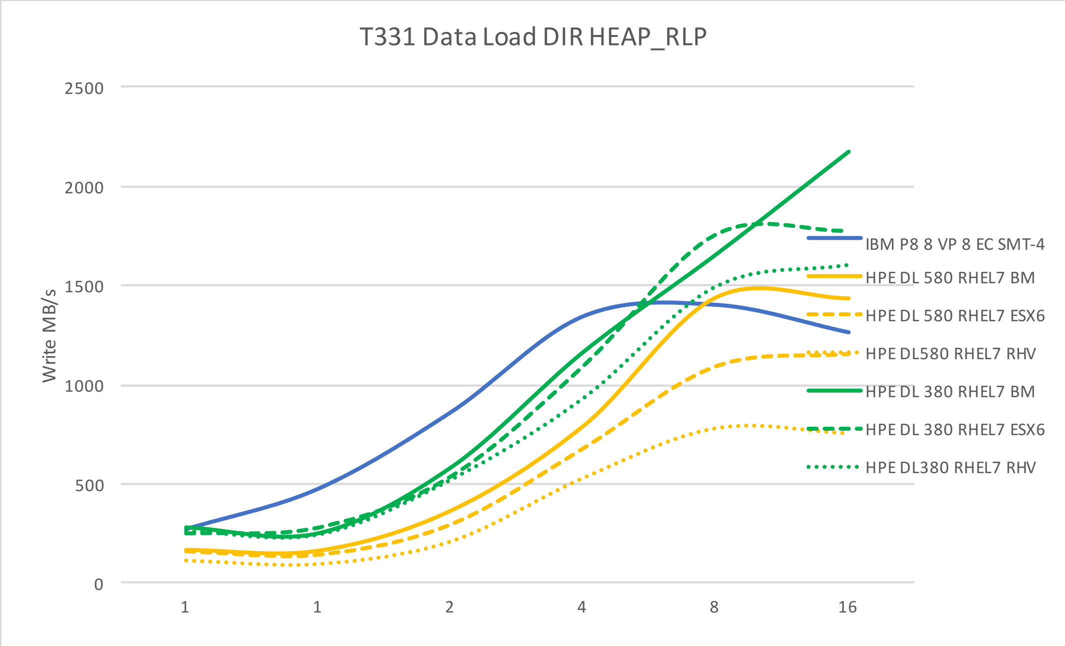

Figure 6 - T331 data load bulk, heap-organised, range-list-partitioned table

With regard to the HPE Intel RHEL systems, this test scales up very well up to a parallelism of 8; at 16 parallel processes, we see no or only a slight increase in throughput. Generally speaking, we can see that the bare metal variant functions very well and that virtualisation causes a poorer throughput. What cannot be fully explained here is the significantly poorer results of the DL580 server, which cannot be explained to this extent by the CPU with slightly lower clock speed. An analysis of the AWR reports showed a significantly higher proportion of the log buffer space wait event. The LGWR process cannot write the data quickly enough. It was possible to reproduce the values when the test was repeated, but unfortunately a further analysis was not possible in the time available.

The IBM systems achieve a significantly poorer throughput in this test, though this is caused in part by the less efficient storage connection. Up to 2 GB/s of I/O are produced in this test. The CPU usage on the HPE Intel RHEL systems with very high throughput is up to 75%, while on the IBM systems only up to 15% of the CPU can be used up.

T332 data load bulk compressed, heap-organised, range-list-partitioned

A bulk insert with a parallelism of 1-16 is carried out with a Commit at the end each time. The test with parallelism 1 is performed twice at the start, so that the total number of inserted rows corresponds to the requirements.

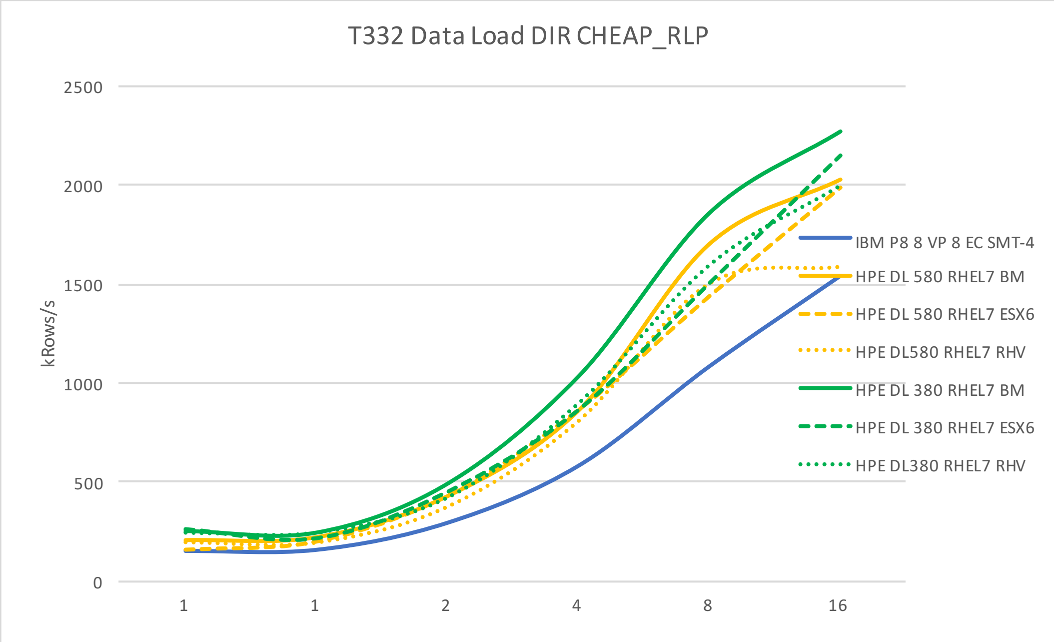

Figure 7 - T332 data load bulk compressed, heap-organised, range-list-partitioned table

In contrast with the T331 tests, in this test all HPE Intel RHEL systems scale up to a parallelism of 16 processes, and even in the virtualised environments there is no significant impact on performance (apart from with RHV on the DL580).

The IBM systems achieve a somewhat poorer throughput in this test. Compared with Test T331, a great deal less needs to be written to the disk due to the compressed heap-organised indexes, so we should not be looking for the cause of the shortfall here. In general, it is probably the case that the IBM platform can be scaled further beyond parallelism 16 due to the 4-way threads.

Summary of T300 data load tests

The T300 load tests, particularly in the conventional loads group (T310/T320), place a very heavy load on the internal mechanisms of the database, most notably the log writer. The T300 bulk tests have a very high I/O proportion.

The table below shows the single thread performance with parallelism 1 (=speed) for the platforms. We see that there are no significant differences in the conventional loads group; with regard to bulk loads, the DL380 systems are the fastest in each case, followed by the DL580 systems and the IBM Power platform.

Tests	Description	IBM P8	DL580 BM	DL580 VM	DL580 RH	DL380 BM	DL380 VM	DL380 RH
T311	Data Load conventional, heap non partioning	6.25	5.95	5.95	6.25	5.68	6.25	5.68
T312*	Data Load conventional, iot non portioning	6.25	5.68	5.68	5.68	6.25	5.68	5.95
T313*	Data Load conventional, hash cluster non portioning	5.95	6.25	3.38	5.68	5.95	5.95	5.95
T321**	Data Load conventional, heap list-partioning	6.25	5.68	6.25	5.95	6.25	6.25	6.25
T322**	Data Load conventional, iot list-partioning	5.95	6.25	5.43	5.68	6.25	5.95	5.68
T331	Data Load bulk direct, uncompressed	210.53	363.64	210.53	326.53	615.38	615.38	615.38
T332*	Data Load bulk, direct, compressed	150.94	210.53	207.79	210.53	253.97	258.06	228.57

Table 6 - T300 data load speed heat map

The table below shows the maximum throughput in 8 physical cores in each case (= throughput). The best throughput is achieved on the DL380 systems, while the DL580 systems show a lower throughput overall, with a traceable effect of virtualisation in each case. The performance of the IBM platform is generally poorer in this respect.

Tests	Description	IBM P8	DL580 BM	DL580 VM	DL580 RH	DL380 BM	DL380 VM	DL380 RH
T311	Data Load conventional, heap non partioning	88.05	85.89	59.32	62.78	166.67	160.92	140
T312*	Data Load conventional, iot non portioning	64.81	37.84	28.28	25.74	95.89	79.55	63.93
T313*	Data Load conventional, hash cluster non portioning	51.47	35.81	26.92	26.67	89.74	75.68	67.31
T321**	Data Load conventional, heap list-partioning	119.66	215.38	186.67	123.89	225.81	229.51	184.21
T322**	Data Load conventional, iot list-partioning	120.69	218.75	186.67	101.45	218.75	208.96	170.73
T331	Data Load bulk direct, uncompressed	1142.86	2245.61	1729.73	1620.25	3240.51	2612.24	2327.27
T332*	Data Load bulk, direct, compressed	1542.17	2031.75	1969.23	1802.82	2265.49	2169.49	1882.35

Table 7 - T300 data load throughput heat map

Results of T400 data load

The tests in the T400 group are effectively the reading counterpart of the writing T200 tests. They simulate a sequential scan of large Oracle tables and are very storage-intensive. Data-scan operations are very common in Oracle (e.g. in table scans, index scans or cluster scans) and are one of the key factors for good Oracle performance.

T426 data scan

This test is a revealing test for the sequential read performance of Oracle. It involves performing a non-cached full table scan with an increasing number of processes. This test forms an important indicator of the efficiency of the entire I/O stack. In each I/O request, 64 8 KB blocks are read, therefore 512 KB per I/O request.

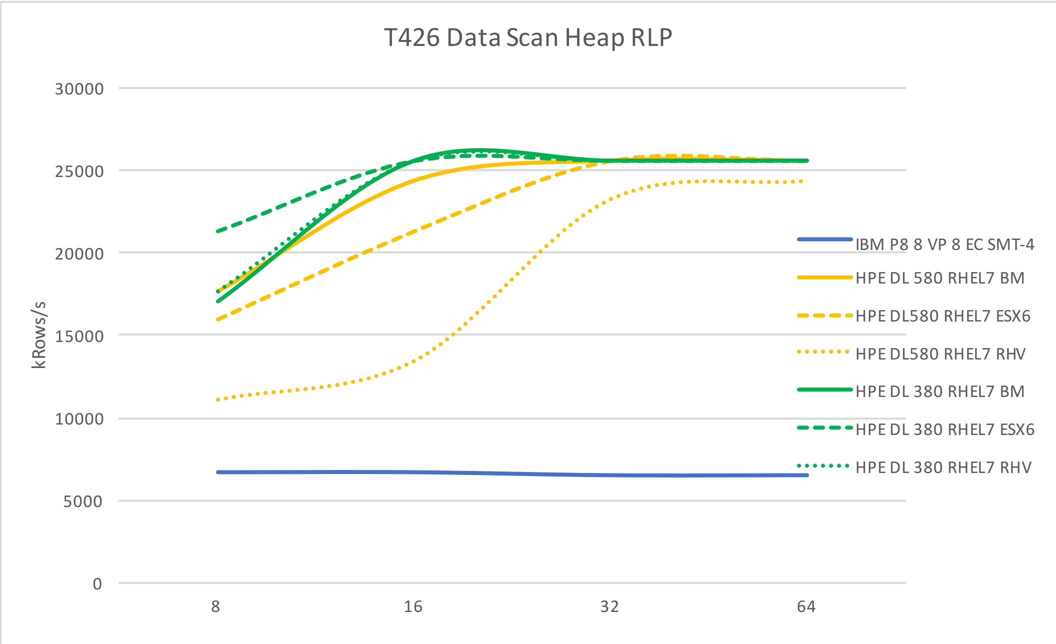

Figure 8 - T426 data scan

The results of the test show that as parallelism increases, the scan rate rises very rapidly to the maximum of just under 6 GB/s for the HPE Intel RHEL systems, while only around 1.6 GB/s are reached on the IBM P8 platform. It must be pointed out, however, that the storage connection of the IBM system was significantly lower at 4 x 8 Gbit (maximum theoretical bandwidth of 4 GB/s) than with the HPE systems at 8 x 16 Gbit (maximum theoretical bandwidth of 16 GB/s). A direct comparison of the data write performance is therefore not permissible; relatively speaking, all systems occupy approximately 35% of the theoretical storage bandwidth.

The CPU performance on the IBM system is in the range of 6-7%, while on the HPE systems it is 30-60% for four times the read performance. This test is essentially I/O-bound.

The important finding of this test is that all Red Hat/HPE platforms are able to scan in Oracle data at almost 6 GB/s and are also able to do this with active virtualisation. These are impressive figures which often creates an important performance advantage in reality.

Results of T500 data aggregation

The tests in the T500 group create aggregations, as they occur in different database applications in both OLTP and in the DWH group, e.g. sorting, grouping, analyses such as AVG, MAX, MIN, etc.).

T512 data aggregation create B-tree (with logging)

In this test, a B-tree index with an increasing number of processes is created. This test forms an important indicator of the efficiency of the entire I/O stack. In each I/O request, 512 KB were read.

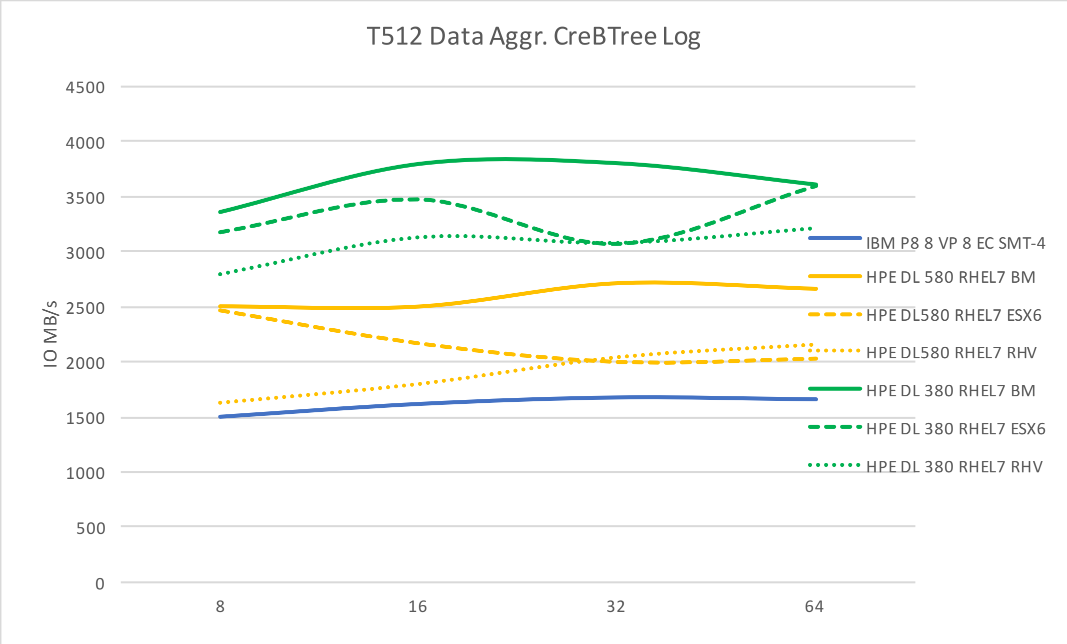

Figure 9 - T512 data aggregation create B-tree index (with logging)

The results of the test show that, even at low parallelism of 8 processes, up to 4 GB/s I/O throughput is generated and this does not increase further with increasing parallelism. Only around 1.6 GB/s throughput can be achieved on the IBM P8 system. It must be pointed out, however, that the storage connection of the IBM system was significantly lower at 4 x 8 Gbit (maximum theoretical bandwidth of 4 GB/s) than with the HPE systems at 8 x 16 Gbit (maximum theoretical bandwidth of 16 GB/s). A direct comparison of the data write performance is therefore not permissible. What is striking is that the virtualised systems deliver significantly lower I/O throughput and the DL580 in turn delivers poorer results than the DL380.

The CPU performance is 20% on the IBM system, and is 60-80% on the HPE systems at higher I/O performance.

Results of T600 data select

The T600 tests simulate random access to individual Oracle database blocks (random select). These operations are highly typical of OLTP applications, but also occur in big data or DWH applications. In principle, a distinction is made between the following tests:

- T610: access via primary key with low selectivity (small hit list)
- T620: access via secondary key with high selectivity (large hit list)

The following six tests are each performed using primary and secondary key access:

- T6x1: heap-organised, non-partitioned, **cached**
- T6x2: index-organised, non-partitioned, **cached**
- T6x3: hash cluster-organised, non-partitioned, **cached**
- T6x4: heap-organised, list partitioned, **cached**
- T6x5: index-organised, list-partitioned, **cached**
- T6x6: heap-organised, range-list-partitioned, **non-cached**

It must be noted that the first 5 tests are all cached; in other words, they work on a very small table that is loaded in the SGA relatively quickly. The storage connection does not play any important role here; rather, it is primarily the processing performance of the DB server that is important. The T616 and T626 tests are non-cached tests on a table that does not fit in the SGA. In these tests, storage latency in particular plays an important role in the processing, and a very large number of I/Os are generated, particularly in test T626 with a large hit list.

T611 data select primary key heap-organised, non-partitioned, cached

Test 611 is shown here as representative for the cached T61x tests. The other tests T612-T615 show similar results. The tests carry out random selects via the primary key, with a very small hit list (one row).

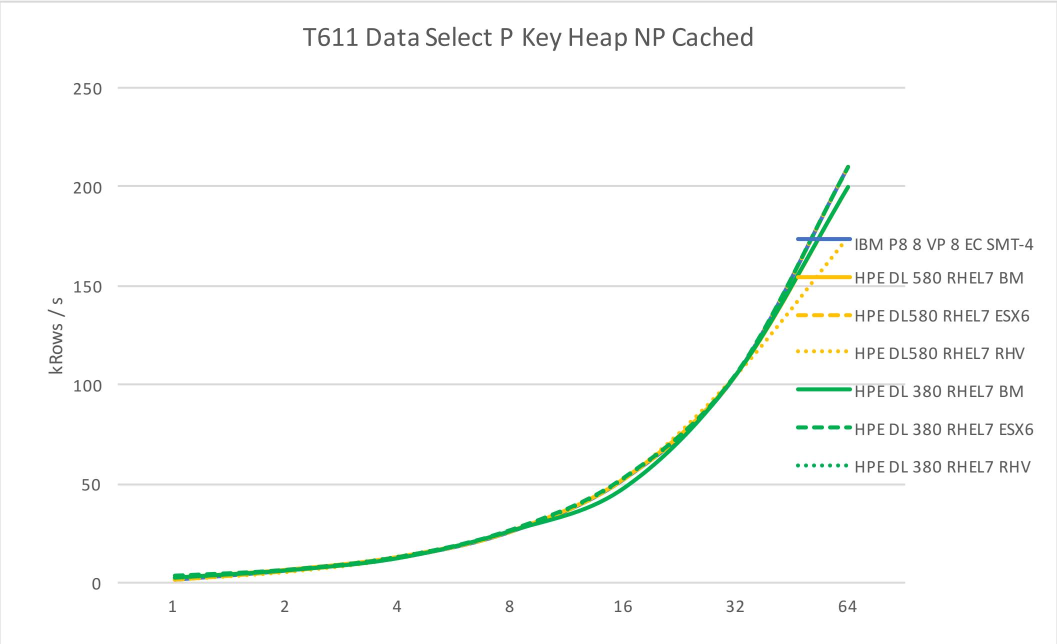

Figure 10 - T611 data select primary key heap-organised, non-partitioned, cached

The results are practically identical for all platforms (the curves sometimes overlap exactly); processing performance of approximately 200,000 rows/s is achieved in each case. The throughput increases continuously up to the maximum chosen parallelism of 64 processes. At 64 processes, the CPU usage was 40% for the IBM server, and 50% for the HPE Intel RHEL servers. It was not possible to increase the maximum processing performance further with additional levels of parallelism. The I/O performance does not play any important role in the cached tests.

T616 data select primary key heap-organised, range-list-partitioned, non-cached

The T616 test performs random selects on a very large table, and consequently the results cannot be cached in a larger style. The hit list of the select is only one row in each case.

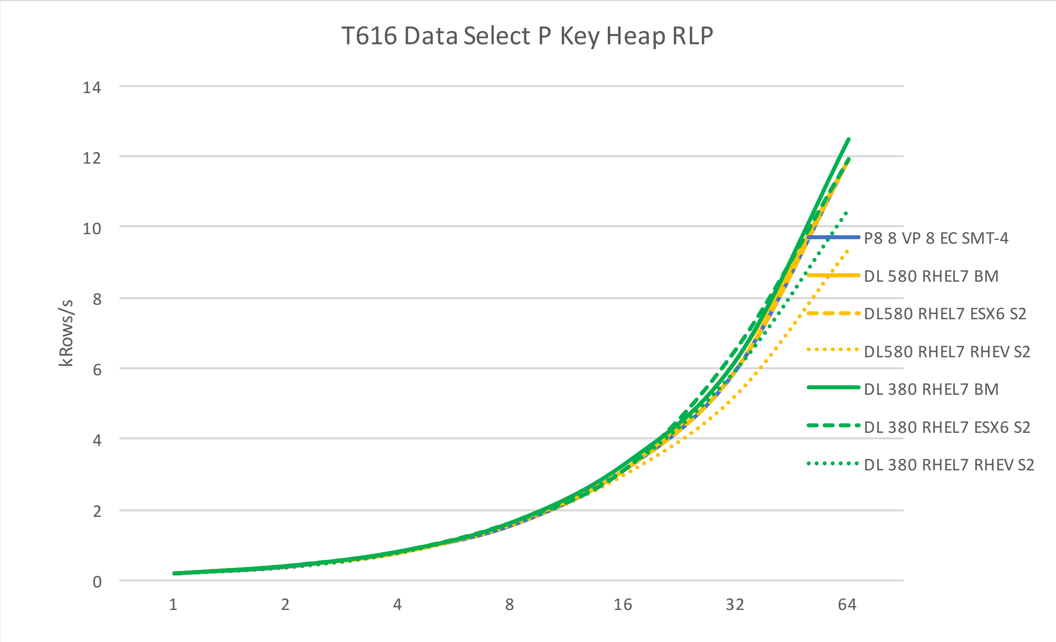

Figure 11 - T616 data select primary key heap-organised, range-list-partitioned, non-cached

The results are very similar for all platforms (the curves sometimes overlap exactly) and processing performance of approximately 9,000 to 12,000 rows/s is achieved in each case. The throughput increases continuously up to the maximum chosen parallelism of 64 processes.

The processing performance on the non-cached table (with approximately 10,000 rows/s) is 20 times lower than for the table in the cache (200,000 rows/s). Here it is possible to see the potential for acceleration of database access in the cache compared with database access from storage (even if this is available as ultra-fast SSD storage).

CPU utilisation was very low in all systems (<20%); storage latency in particular plays an important role in this test. According to the AWR report, this was approximately 0.5 ms in all tested configurations.

T621 data select secondary key heap-organised, non-partitioned, cached

Test 621 is shown here as representative for the T62x tests. The tests T622-T625 show similar results. The tests perform random selects via the secondary key, which deliver a large number of hits.

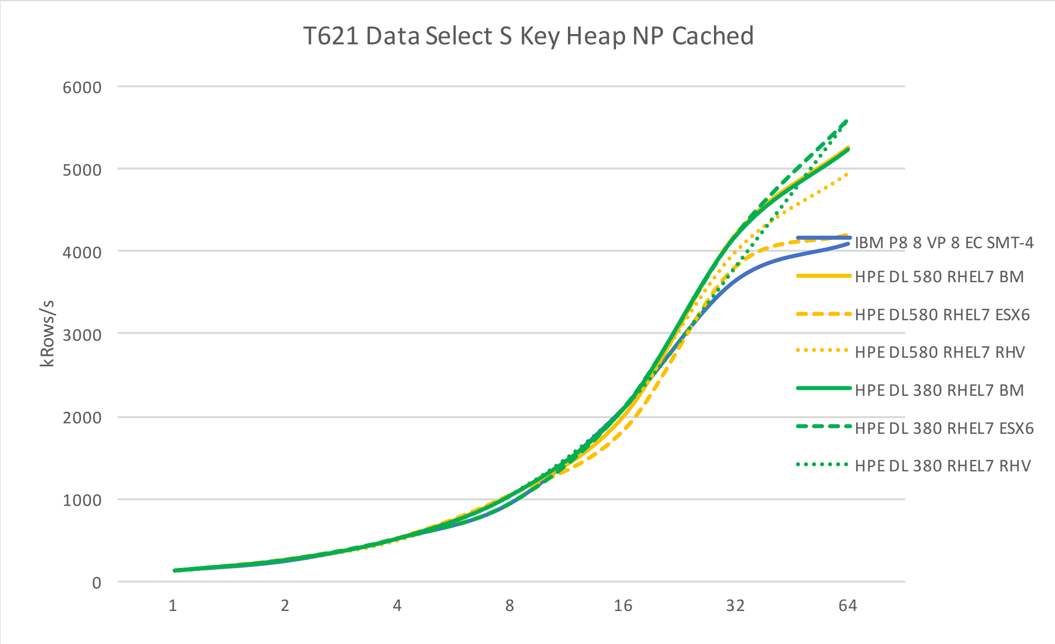

Figure 12 - T621 data select secondary key heap-organised, non-partitioned, cached

The results are similar for all platforms; processing performance of 4 to 5.6 million rows per second was achieved in each case. The throughput increases continuously up to the maximum chosen parallelism of 64 processes. Due to the larger hit list, this test is very CPU-intensive. In the HPE Intel RHEL systems, the CPU usage was already almost 100% at 64 processes, while in the IBM system it was 75%. The type of virtualisation on the HPE Intel RHEL servers does not play any determining role. The I/O performance does not play any important role in the tests.

T626 data select secondary key heap-organised, range-list-partitioned, non-cached

Test T626 performs random selects on a very large table, as a result of which the results cannot be cached. The hit list of the select is large in each case and the I/O performance therefore plays a major role.

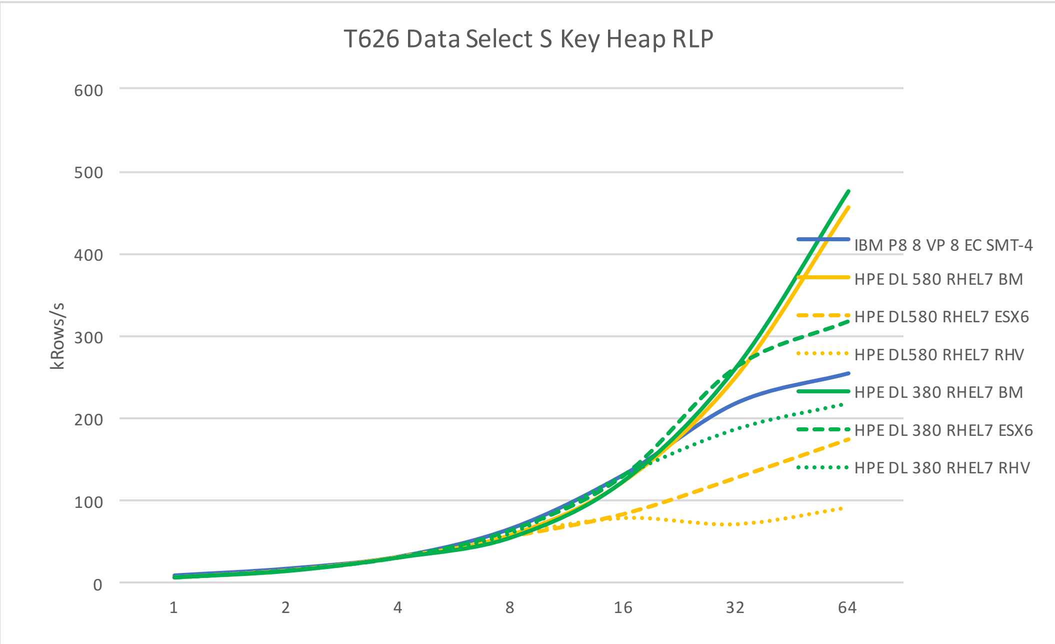

Figure 13 - T626 data select secondary key heap-organised, range-list-partitioned, non-cached

The results are relatively clearly distinguished in these tests. In particular, we see a clear effect of virtualisation on the HPE Intel RHEL systems, which evidently are unable to handle a very large number of small I/Os very efficiently and in parallel. This is especially the case in the set-up of the DL580 systems. On the other hand, the bare metal results of the x86 servers are excellent and demonstrate twice the throughput of the IBM platform.

The analysis of the results shows that the CPU usage of the HPE systems is up to 80% (for bare metal) and, logically, is significantly lower for the virtualised environments with lower throughput. The analysis of the service times shows that these correlate directly with the throughput. For the bare metal systems, the average service time of user I/Os is less than 10 ms, while this doubles to 20 ms on the DL380 with RHV and even more than quadruples to 46 ms on the DL580 with RHV. The IBM Power platform lies in the middle with a service time of 17 ms.

The processing performance on the non-cached table (maximum of 500,000 rows/s) is 10 times lower than for a table in the cache (with 5 million rows/s). Here we see the potential for acceleration of database access in the cache compared with database access in the storage (even if this is available as ultra-fast SSD storage).

Summary of T600 data select tests

The table below shows the single thread performance with parallelism 1 (=speed) for the platforms. In most tests, there are no significant differences in the case of one individual thread. In particular, the non-cached tests 616 and 626 show identical results with 1 thread.

Tests	Description	IBM P8	DL580 BM	DL580 VM	DL580 RV	DL380 BM	DL380 VM	DL380 RV
T611	Data select, random via primary key, heap non partioning, cached	1.93	2.52	1.93	1.56	2.52	2.98	2.43
T612*	Data select, random via primary key, iot non partioning, cached	3.28	3.28	3.28	3.28	3.28	3.12	3.28
T613*	Data select, random via primary key, hash cluster non partioning, cached	2.05	2.11	2.85	1.93	2.62	2.98	2.11
T614*	Data select, random via primary key, heap list portioning, cached	2.85	2.43	1.04	0.68	2.34	1.46	2.62
T615*	Data select, random via primary key, iot list partioning, cached	3.28	3.28	3.28	3.28	3.28	3.28	3.28
T616	Data select, random via primary key, non cached	0.01	0.01	0.01	0.01	0.01	0.01	0.01
T621*	Data select, random via secondary key, non-partionied, cached	131	131	125	131	131	131	131
T622*	Data select, random via secondary key, iot non partioning, cached	131	125	105	105	131	131	131
T624*	Data select, random via secondary key, heap list portioning, cached	0.82	0.71	0.82	0.82	0.82	0.78	0.82
T625*	Data select, random via secondary key, iot list partioning, cached	131	114	131	131	131	125	131
T626	Data select, random via secondary key, non cached	8.36	8.25	8.22	8.06	8.34	8.19	8.06

Table 8 - T600 data load select heat map

The table below shows the maximum throughput in 8 physical cores in each case (= throughput). Only relatively small differences could be seen in the cached tests 611-615 and 621-625. In the non-cached tests T616 and T626, on the other hand, a traceable effect of virtualisation could be seen. The results for the IBM systems lie in the middle range in these tests.

Tests	Description	IBM P8	DL580 BM	DL580 VM	DL580 RV	DL380 BM	DL380 VM	DL380 RV
T611	Data select, random via primary key, heap non partioning, cached	210	200	210	175	200	210	210
T612*	Data select, random via primary key, iot non partioning, cached	200	210	210	210	210	210	210
T613*	Data select, random via primary key, hash cluster non partioning, cached	200	210	210	210	210	210	210
T614*	Data select, random via primary key, heap list portioning, cached	210	210	210	210	210	210	210
T615*	Data select, random via primary key, iot list partioning, cached	210	210	200	210	210	210	210
T616	Data select, random via primary key, non cached	11.92	11.92	11.92	9.36	12.48	11.92	10.49
T621*	Data select, random via secondary key, non-partionied, cached	4'092	5'243	4'195	4'935	5'243	5'592	5'592
T622*	Data select, random via secondary key, iot non partioning, cached	3'495	3'355	2'943	3'165	2'996	3'570	3'647
T624*	Data select, random via secondary key, heap list portioning, cached	4'794	5'084	4'661	4'935	5'992	5'785	5'786
T625*	Data select, random via secondary key, iot list partioning, cached	4'194	4'302	3'813	3'995	4'415	4'660	3'814
T626	Data select, random via secondary key, non cached	255	456	174	92	477	318	219

Table 9 - T600 data select throughput heat map

Results of T700 data update

The T700 tests simulate an update of individual database elements (random update). These operations are highly typical of OLTP applications, but also occur in big data or DWH applications.

The following 6 tests were performed, each with primary key access and low selectivity (1 row):

- T711: heap-organised, non-partitioned, **cached**
- T712: index-organised, non-partitioned, **cached**
- T713: hash cluster-organised, non-partitioned, **cached**
- T714: heap-organised, list partitioned, **cached**
- T715: index-organised, list-partitioned, **cached**
- T716: heap-organised, range-list-partitioned, **non-cached**

It must be noted that the first 5 tests are all cached; in other words, they work on a very small table that is loaded in the SGA relatively quickly. The storage connection does not play any important role here; rather, it is primarily the processing performance of the DB server that is important. Test T716 is a non-cached test on a table that does not fit in the SGA. In this test, the storage latency in particular plays an important role for the processing, and many asynchronous block updates and redo log entries are also generated.

T711 data update primary key heap-organised, non-partitioned, cached

Test T711 is shown as representative of the T71x tests. The tests T712-T715 show similar results. The tests perform random updates via the primary key, which deliver a very small hit list (one row).

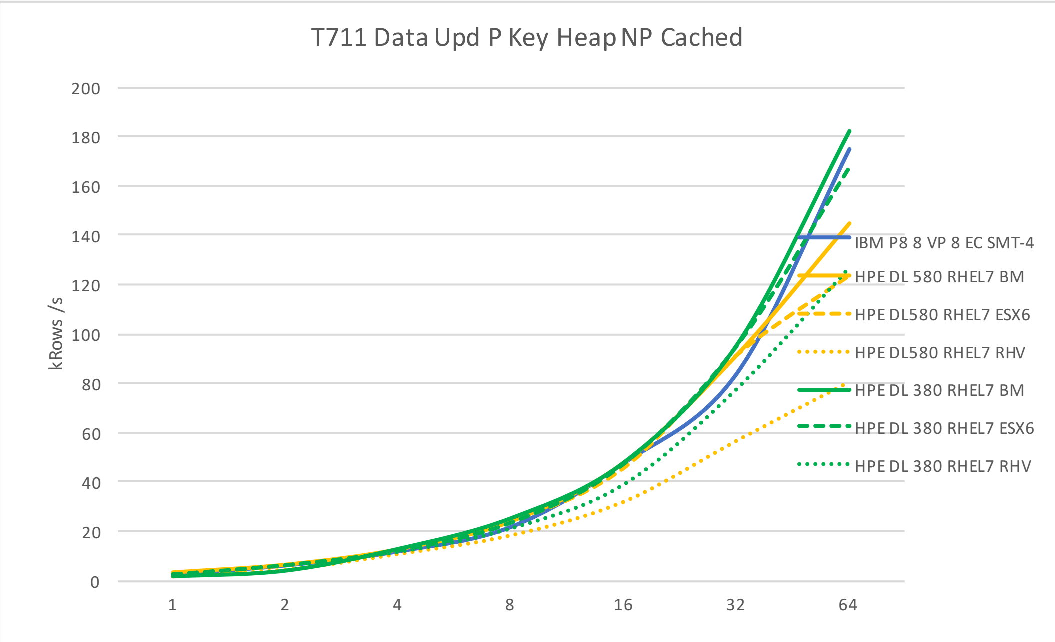

Figure 14 - T711 data update primary key heap-organised, non-partitioned, cached

The results are relatively clearly distinguished in these tests. In particular, we see a clear effect of the Red Hat virtualisation RHV on the HPE Intel RHEL systems, which evidently are unable to handle a large number of small I/Os very efficiently and in parallel. This is especially the case in the set-up of the DL580 systems. The bare metal results of the x86 servers are comparable to the throughput of the IBM P8 platform.

The analysis of the results shows that maximum CPU usage is 80% (for bare metal) and is accordingly lower for the virtualised environments with lower throughput. Otherwise, the essential difference in relation to the virtualised environments lies in greater wait times for the log buffer space event; in other words, the redo logs cannot be saved as quickly in these systems, which reduces throughput. This wait event does not arise with IBM, and this is why the results in this test are especially favourable compared with the other results.

This test is quite storage-intensive, as up to 20,000 IOPS and up to 600 MB/s are written (redo logs). The number of reading I/Os is accordingly low due to the caching.

T716 data update primary key heap-organised, range-list-partitioned, non-cached

Test T716 performs random updates on a very large table, and consequently the results cannot be cached in a larger style. The hit list of the select is only one row in each case.

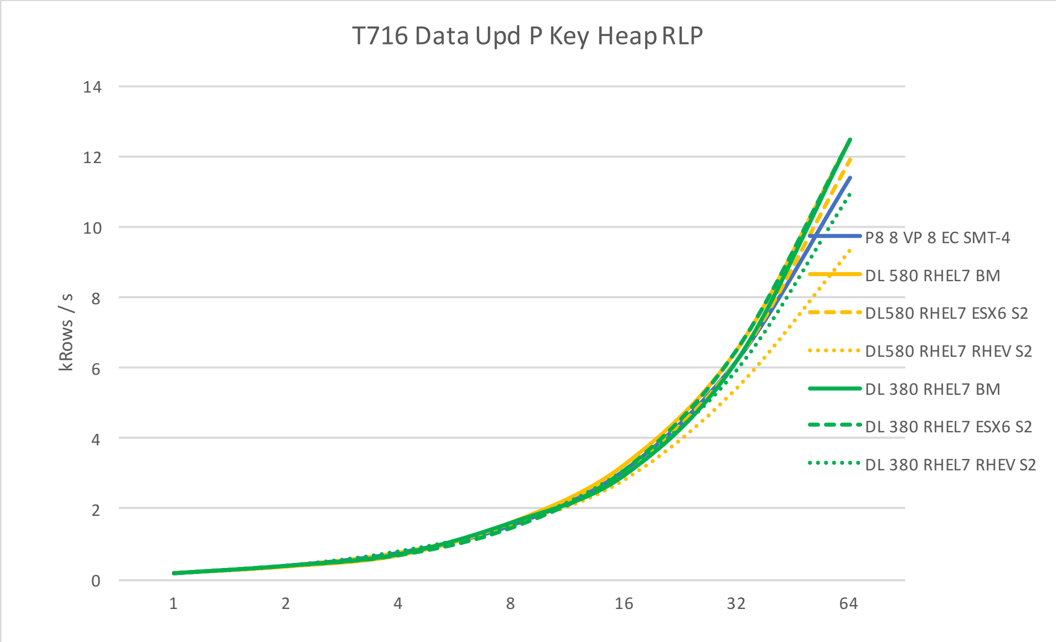

Figure 15 - T716 data update primary key heap-organised, range-list-partitioned, non-cached

The results are very similar for all platforms (the curves sometimes overlap exactly) and processing performance of approximately 9,000 to 12,000 rows/s is achieved in each case. The 2 environments virtualised with RHV have a minimal throughput. The throughput increases continuously up to the maximum chosen parallelism of 64 processes.

The processing performance on the non-cached table (approximately 10,000 rows/s) is 20 times less than that of the small table in the cache (with 180,000 rows/s). Here it is possible to see the potential for acceleration of database access in the cache compared with database access from storage (even if this is available as ultra-fast SSD storage).

CPU utilisation was very low in all systems (<20%); storage latency in particular plays an important role in this test. According to the AWR report, this was approximately 0.5 ms in all tested configurations.

Summary of T700 data update tests

The table below shows the single thread performance with parallelism 1 (=speed) for the platforms. There are no differences in the non-cached Test 716. In the other tests, the virtualised environments are generally somewhat slower than the bare metal environments. The IBM P8 platform is on a par with the HPE Intel RHEL servers in these tests.

Tests	Description	IBM P8	DL580 BM	DL580 VM	DL580 RV	DL380 BM	DL380 VM	DL380 RV
T711	Data update, random via primary key, heap non partioned, cached	2.62	3.12	1.72	1.37	1.29	2.85	2.26
T712*	Data update, random via primary key, iop non partioned, cached	1.26	1.46	1.31	0.91	2.98	1.87	1.52
T713*	Data update, random via primary key, hash custer non partioned, cached	2.62	2.26	2.98	2.11	1.99	2.18	1.93
T714*	Data update, random via primary key, heap list-partioned, cached	1.04	0.95	0.9	0.64	2.62	1.26	1.06
T715*	Data update, random via primary key, iot list-partioned, cached	1.31	1.42	1.34	1.04	2.11	1.99	1.68
T716	Data update, random via primary key, range-list partioned, non cached	0.2	0.2	0.2	0.2	0.2	0.2	0.2

Table 10 - T700 data load select heat map

The table below shows the maximum throughput in physical cores in each case (= throughput). No differences are evident in the non-cached Test 716. In the other tests, the virtualised environments are generally somewhat slower than the bare metal environments. In these tests, the IBM is on a par with the HPE Intel RHEL servers.

Tests	Description	IBM P8	DL580 BM	DL580 VM	DL580 RV	DL380 BM	DL380 VM	DL380 RV
T711	Data update, random via primary key, heap non partioned, cached	175	145	123	81	182	168	127
T712*	Data update, random via primary key, iop non partioned, cached	168	140	123	82	161	145	108
T713*	Data update, random via primary key, hash custer non partioned, cached	175	155	145	84	191	175	131
T714*	Data update, random via primary key, heap list-partioned, cached	168	155	131	84	182	145	127
T715*	Data update, random via primary key, iot list-partioned, cached	120	117	108	84	155	150	98
T716	Data update, random via primary key, range-list partioned, non cached	11.4	12.48	11.92	9.36	12.48	12.48	10.92

Table 11 - T700 data select throughput heat map

Results of T800

The tests in the T800 group perform data joins in a nested loop.

T816 data join nested loop, heap-organised, range-list partitioned, non-cached

In this test, a data join is created with an increasing number of processes.

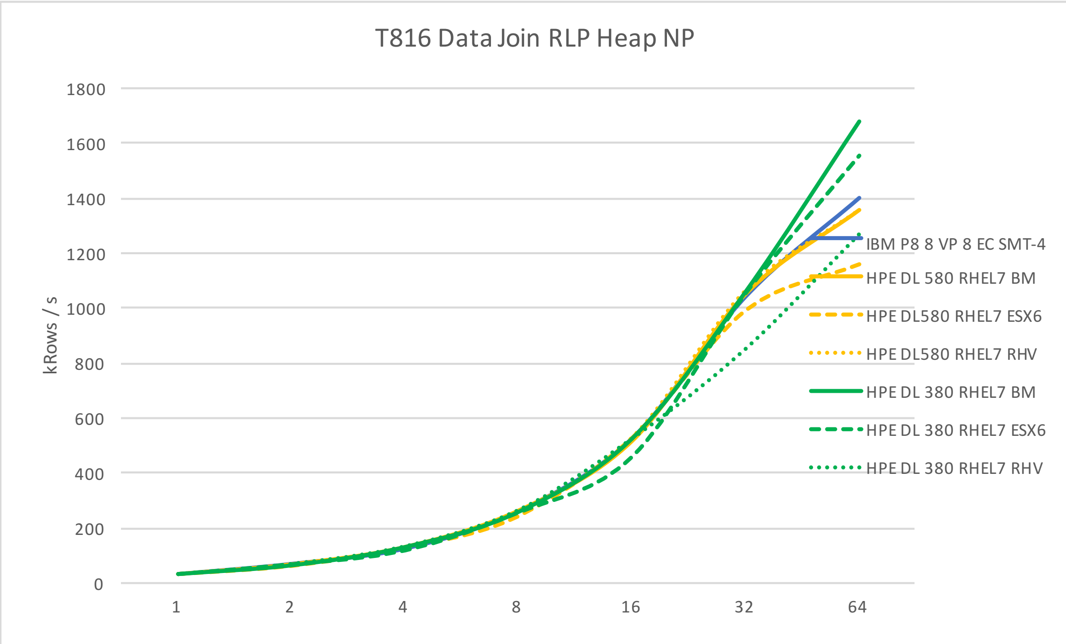

Figure 16 - T816 data join nested loop, heap-organised, range-list partitioned, non-cached

The results are relatively similar on all tested platforms, particularly at a low parallelism up to 16 processes. At high parallelisms, the systems are very heavily loaded (CPU usage > 90%) and the I/O performance is also relatively high at over 40,000 random reads per second.

Summary

The CPU-intensive T100 tests show the HPE Intel RHEL systems to have a clear advantage. This is especially true of the single thread performance (speed) but also for overall throughput. With the same number of threads, the HPE Intel RHEL systems are markedly faster, while the IBM platforms generally require a higher parallelism in order to reach their maximum performance. The effect of virtualisation in the CPU-intensive tests is not significant, and the same performance is achieved with virtualisation as without it (bare metal). The E5-2637 v4 processor with 3.5 GHz clock speed that is built into the DL380 is noticeably faster than the E7-8893 v4 processor with 3.2 GHz clock speed that is used in the DL580. In this test with only 8 physical cores, the DL380 system therefore demonstrates the higher throughput; for reasons of comparability, only half of the available 16 cores in the DL580 were used.

The storage-intensive T200 and T400 tests were not truly comparable due to the different type of storage connection used by IBM and HPE. However, we see a consistently excellent I/O rate of up to 5 GB/s both for sequential writing and for sequential reading on all HPE platforms. These values were also achieved in the virtualised environments.

In load scenarios (T300) and for data aggregation (T500), the HPE Intel RHEL servers are, without exception, better than the Power platform, though a slightly greater performance impact from virtualisation could be measured.

In the data selects (T600), the results were generally comparable; in the cached tests, they were limited by the processing performance of the Oracle database, while in the non-cached tests, they were limited by the service time of the storage systems. Test T626 formed the exception; this showed a much greater impact of virtualisation, while the bare metal results of the HPE Intel RHEL systems were significantly better than the IBM P8 results.

Data update (T700) and data join (T800) show similar performance on all systems, but likewise with a traceable effect of virtualisation on the HPE Intel RHEL servers.

Conclusion

The CPU performance of contemporary HPE servers with current Intel processors using RHEL has now surpassed the contemporary Power 8 systems using AIX, and this is especially apparent in the Oracle PL/SQL benchmarks.

The tests with very high I/O throughput are not genuinely comparable between IBM and HPE Intel RHEL, because the I/O connection of these servers was considerably higher in the test than that of the IBM systems. However, it is evident that very high data writes and data scan rates can be achieved on all HPE Intel RHEL systems, which, with 5 GB/s, would be sufficient for very high requirements.

The HPE Intel RHEL stack also demonstrates very good performance in other tests, which exceeded the performance of the IBM platform in practically all tests. In a number of tests, however, a significant performance impact was evident as a result of virtualisation, and this was especially high for a relatively large number of parallel small I/Os. Generally speaking, VMware ESX has a lower impact in this respect than the Red Hat virtualisation RHV.

In conclusion, the use of Oracle databases on x86 systems with Red Hat Linux is recommended from a performance perspective. This is particularly true in view of the fact that performance is higher per core, but only half the licence fees are incurred per Intel core (licence factor 0.5) compared with Power cores (licence factor 1.0).

We must remember that IBM Power platforms offer excellent virtualisation, and cope very well even with very high CPU loads up to 80% and with a very large number of virtual systems. In contrast, loads of up to around 50% are usually worked with on x86 systems with VMware or RHV. However, the hardware costs for Intel systems are several times lower than those of the Power systems.

The aim of this benchmark is to perform a comparison based on 8 physical cores. This involved measuring 2 different Intel CPUs: the E5 family (maximum of 2 sockets) and the E7 family (2 to 16 sockets).