

System performance measurements, IO and CPU (Orion and jloci)

The goal of this page is to share the results of the database system performance specs among the 3D installations In particular 2 simple benchmark tools are used for comparison: Oracle's Orion for IO subsystem speed and the jloci test to measure CPU-to-memory speed. See below for further details on how to use those tools. Please note that what is reported below are somewhat 'synthetic' benchmark numbers, so they should be taken with a grain of salt!

IO Results

Storage type and location	System details	server HBA	random small IO, IOPS	Scalability factor	Comments
It rac CERN	128 SATA disks in 8 disk arrays, infortrend 1GB cache, JBOD config, FC 4Gbps	4Gbps	12000	~100 IOPS per SATA disk with destorking	measured with Orion and confirmed with Oracle SQL using a test table of 20GB
dualcore cms	96 Raptor disks in 6 disk arrays, infortrend 1GB cache, JBOD config, FC 4Gbps	4Gbps	16000	~160 IOPS per Raptor disk	measured with with Oracle SQL using a test table of 80GB, note Orion could not be stabilized for this test
PICS	80 SAS disks, Netapps SAN with RAID-DP (RAID6) , 8 GB cache, Controller FAS 3040 , 4Gbps	4Gbps	17000	~210 IOPS per SAS disk	measured with Orion
TRIUMF	9 SATA disks, HP Storage Works	FC	550 (to be confirmed)	-	measured with Orion

- Please see below for the details on how to use Orion to measure IO numbers, in particular the small random IOPS (Orion will measure the maximum IOPS obtained 'at saturation' by submitting hundreds of concurrent async IO requests of 8KB blocks). Sequential IO performance is almost inevitably the HBA speed, that is typically 400 MB per sec, or 800 MB when multipathing is used.

CPU measurements and results

DB server type and location	System details	jloci run time	Estimated speedup	Comments
it rac CERN	dual Xeon@3GHz, 4GB DDR2 @400 MHz, RHEL 4 U5 64 bit, Oracle 10.2.0.3	28 sec	1.0	RAC2,3,4
quadcore cms	dual Xeon 5130@2GHz, 4MB L2, 8 GB FB-DIMM @666MHz, RHEL4 U5 64 bit, Oracle 10.2.0.3	16 sec	7.0	Dell PowerEdge 2950, cms online
quadcore IT CERN	dual Xeon E5345@2NOSPAMPLEASE.33GHz, 8MB L2, 16 GB FB-DIMM @666MHz, RHEL4 U5 64 bit, Oracle 10.2.0.3	20 sec	5.6	test server, 2007
quadcore IT CERN	dual Xeon E5410@2NOSPAMPLEASE.33GHz, 12MB L2, 16 GB FB-DIMM @666MHz, RHEL4 U5 64 bit, Oracle 10.2.0.3	10.6 sec	10.6	Dell PowerEdge 2950, RAC5,6

- **How should I measure CPU performance?** Use the jloci test to measure CPU-to-memory access throughput (single threaded): jloci
- **I would like to run also CPU performance tests, would should I use?** A good starting point is a test of the logical IO speed (memory access), you can find the script here: <https://twiki.cern.ch/twiki/pub/PSSGroup/QuadCoreTests/jloci.sql>
- **What is 'estimated speedup'?** Estimated speedup is a calculated value: single thread speed up measured with the jloci test * number of cores and normalized so to make speedup = 1 on the 'legacy' CERN itrac servers.

Orion testing FAQs

- **What is Oracle Orion:** A small utility distributed by Oracle to measure IO subsystem performance
- **Can I measure IO performance directly using Oracle:** yes, but it's more complex, see some examples and SQL code in HAandPerf
- **When should I use Orion:** Ideally when installing a new system before installing Oracle
- **Do I need the Oracle RDBMS or ASM to run Orion?** No
- **Where do I find Orion?:** From Oracle's OTN: Orion on OTN [OTN](#), download orion_linux_em64t.gz (11g version for x86_64)
- **Can I run Orion on a system with Oracle installed and running?** You shouldn't, anyway it will run. In this case you should make absolutely sure that you run Orion with the option -write 0 (default) and accept the risk of the impact on your service levels to have an overloaded IO subsystem.
- **How long does it run for:** it depends on the parameters, for a simple run a typical duration is ~1 hour
- **What are the main parameters I should use:** num_disks to match the number of physical disks, cache_size to match your IO (array) cache size.
- **What is a first simple run that I can use to test Orion:** create a file mytest.lun with 2 LUNS (one per line, say /dev/sdb1 and /dev/sdc1), then run as root ./orion_linux_em64t -run simple -testname mytest -num_disks 2
- **A more complete command line for extensive tests:**
 - ◆ attach all the storage as you would do for an Oracle installation with ASM, i.e. make visible all the LUNs/disks visible as /dev/sd.. block devices in Linux or if you are using multipath as /dev/mapper/..
 - ◆ create partitions on those LUNs as you would do for ASM setup (for example 1 partitions spanning the whole disk or more partitions to optimize IO access with destroking)
 - ◆ create a file with the list of partitions to be used for the test. Each partition path in a separate line. Example:


```

          ◇ vi mytest.lun and then insert N. lines, one per disk:
          ◇ /dev/sdb1
          ◇ /dev/sdc1 .. etc (let's say 32 lines for a 32-disk test)
          
```
 - ◆ Run the test as root, example: ./orion_linux_em64t -run advanced -write 0 -matrix basic -duration 120 -testname orion_test3 -simulate raid0 -num_disks 32 -cache_size 4000

How to read Orion output and common gotchas

- The summary file for a simple run you will produce 3 numbers: Maximum Large MBPS, Maximum Small IOPS, Minimum Small Latency
- Maximum MBPS typically saturates to the HBA speed. For a single ported 4Gbps HBA you will see something less than 400 MBPS. If the HBA is dual ported and you are using multipathing the number should be close to 800 MBPS
- IOPS is the most critical number. That's is the measurement of the max number of small IO (8KB, i.e. 1 Oracle block) operations per second that the IO subsystem can sustain. It is similar to what is needed for a OLTP-like workload in Oracle, although Orion uses async IO for this tests unlike typical RDBMS operations)
- The storage array cache can play a very important role in producing bogus results (tested). The parameter -cache_size in Orion tests should be set appropriately (in MB). If you can make a test with

the array cache disabled.

- When running read-only tests on a new system an optimization can kick in where unformatted blocks are read very quickly. I advise to run a `at least one write-only test` (that is with `-write 100`) on a new system.

This topic: PSSGroup > OrionTests

Topic revision: r13 - 2008-09-25 - LucaCanali



Copyright &© 2008-2021 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

or Ideas, requests, problems regarding TWiki? use [Discourse](#) or [Send feedback](#)