ORACLE
DATABASE

Oracle White Paper
September 2010

Oracle Database Optimization for MSC SimManager using Oracle's Sun FlashFire Technology

ORACLE

Introduction

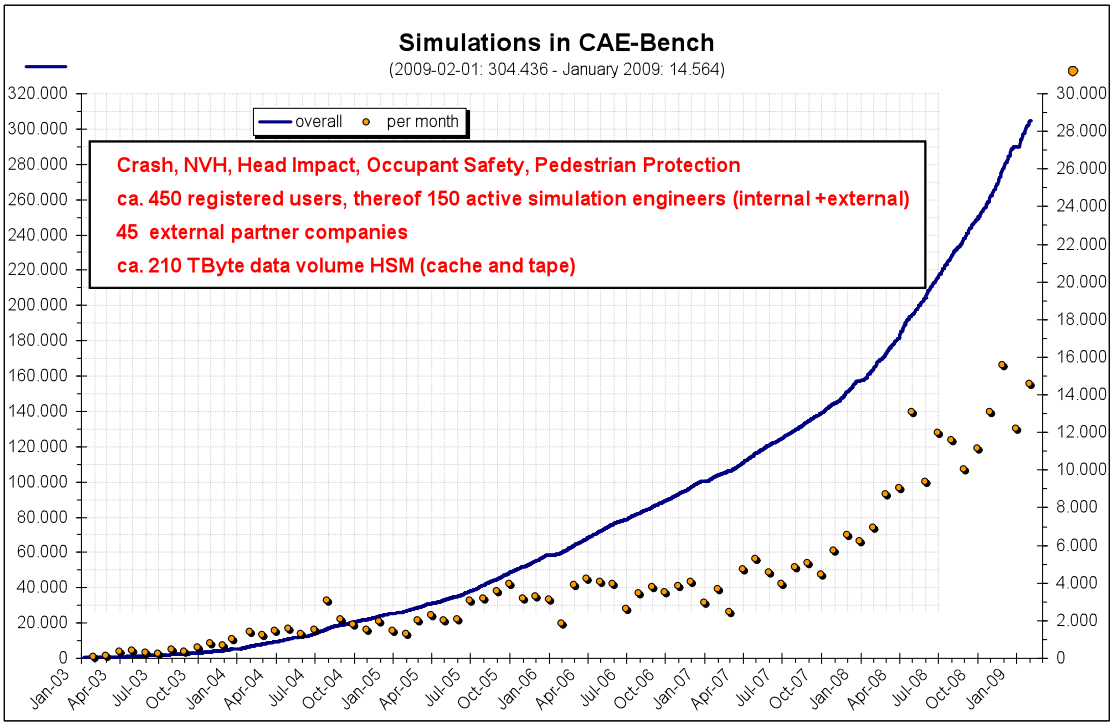
During past decades simulation has become an indispensable tool for manufacturers worldwide to improve their products, for example in Aerospace, Automotive, Shipbuilding, electronics, consumer goods, and other industries. However, the growing amount, size and complexity of simulations lead to a huge amount of simulation data and processes. Today, these data and processes are business critical assets, to be available in real-time, enterprise wide, 24x7. For this purpose, simulation management systems have been developed for managing simulation data and processes creating this data. These systems are usually based on a multi-tier architecture with business logic, database and file vaults to allow efficient management of even the highest volumes of data. For this purpose, simulation management systems such as MSC SimManager have been developed. For further information on MSC SimManager please see: http://www.mscsoftware.com/Products/Virtual-Build-And-Test-Management/SimManager.aspx. As shown in this paper, MSC SimManager utilizes the Oracle database for efficient management of meta-data.

Like simulation codes themselves, such simulation management systems pose high requirements on underlying IT infrastructures, due to the enormous amounts of data to be processed. Storage and database systems play a crucial role in enabling efficient management of simulations. This paper will provide best practice configuration guidelines for using Oracle's Sun FlashFire technology storage devices along with Oracle's Sun servers to optimize simulation management solutions.

As an example, the automotive industry faces the challenge that an ever increasing amount of car lines and variants of car lines are needed to be successful in both mature and emerging markets. Legal requirements, customer demands for quality, comfort and safety in parallel to shorter development cycles and tight budgets result in more simulations with higher complexity. In order to optimize the simulation processes, the automotive industry has been a leading adopter of commercial simulation management

systems. Not only data is managed, but also the processes to create the data are standardized and automated, to increase throughput, quality and reliability of simulation.

The resulting amount of data to be managed is huge, as shown in the statistics below (from: "CAE Process and Data Management at Audi", Karl Gruber, Josef Reicheneder, Stefan Bauer, Manfred Keul, AUDI AG Ingolstadt, NAFEMS Seminar: „Simulation Data Management - Integration into the Product Development Process", Wiesbaden, March 2009). The simulation management system at Audi is called CAE-Bench and is based on MSC SimManager. In February 2009, the system contained already more than 300,000 simulations. Each simulation results in an average of about 200 data objects to be managed such as submodels, load data, input files, results, post processing objects, reports, and process data, and can have gigabytes of raw data associated with it. This gives a good impression of the amount of database and file sizes a simulation management system has to cope with.



Simulations in MSC SimManager Deployment at Audi ("CAE-Bench") per month and accumulated

As outlined before, this paper will provide best practice configuration guidelines for using latest storage and server technology (Oracle's Sun FlashFire technology storage devices along with Oracle's Sun servers) to optimize such simulation management solutions. Using the example above, the corresponding database would contain about 70 million objects. Due to the continuous increase of using simulations, a growth of up to 100% of the database per year has been observed in customers using simulation management solutions, which requires us to set the stage now regarding how to cope with such data volumes in the near future.

In our tests with simulation management databases up to 235 million objects we used Oracle's Sun FlashFire technology and best practice system tuning techniques to drastically remove I/O bottlenecks for typical simulation management user activities.

Subsequent sections of this paper will show how we achieved up to 5x speedups through acceleration of Oracle's 11gR2 Database and application performance being run concurrently on a highly integrated well balanced system.

The paper describes the results of research done in this area in collaboration with MSC.Software, a leading provider of simulation management systems that utilize the Oracle Database in industries around the world. MSC's customers who have adopted its simulation management solution MSC SimManager include the Aerospace, Automotive, Shipbuilding, consumer goods, and other industries. Sun and Oracle have worked closely with MSC for over 20 years to provide best practice hardware and software solutions to its customers.

The key points this paper will address are:

1.) Techniques for accelerating performance [faster response times] on large database configurations with Oracle's Sun Flash storage (such as the MSC SimManager generated **235 million object, 500gb database** used for the use cases in this paper).

    This paper will describe best practice guidelines for improving response times for Oracle database access by MSC SimManager. MSC.SimManager is a simulation management system that uses the Oracle database to store, manage, and later reference information (meta-data) about each simulation run made by an organization. Since the database grows proportional to the number of simulations it becomes increasingly important to have methods to maintain fast database access and retrieval response times as the database grows. Engineering organizations are now able to generate and store hundreds of thousands of simulations and hundreds of millions of database objects through the automation and improved productivity provided by the simulation management systems of MSC.SimManager. With these database sizes conventional database tuning solutions such as storing the database within physical memory are either costly or may not be feasible in every situation.

This paper will address alternative performance improvement solutions available through the use of Oracle's Sun FlashFire technology.

2.) Techniques for Improving job throughput—"More jobs in less time".
Best practice recommendations are provided for maximizing an engineering organizations simulation job throughput on the new high core count compute servers using Oracle's Sun FlashFire technology. Examples will be shown that illustrate this improved job throughput (e.g., 50% more simulations being run with less than 20% elapsed time overhead per job)

.

The I/O Bottleneck

Processor performance, especially in high-performance clustered multiprocessor systems, has grown much more quickly than the performance of I/O systems and large-scale storage devices. At the same time, high-performance computing tasks in particular have been dominated more and more by the need to manage and manipulate large data files, such as files containing sensor data for meteorology and climate models. In combination, the need to manage these large data files while meeting the data demands of fast processors has led to a growing imbalance between computation and I/O — the *I/O Bottleneck*. This I/O Bottleneck has also become a serious challenge and limitation for database applications that are attempting to help manufacturing companies bring their products to market more quickly. These database applications frequently cope with huge volumes of corporate information assets. While the demand for capacity continues to skyrocket, database performance tends to correlate to how quickly data can be accessed and retrieved from storage for processing. Unfortunately, many currently available database system architectures suffer from bottlenecks between storage components and database servers. Typically the bottlenecks are related to restricted I/O bandwidth and how fast random I/O operations can occur relative to how many input/output operations per second (IOPS) the storage can provide. It therefore has become essential to look for performance improvements somewhere other than increased processor speed that also focus on the IOPS being provided to a given system.

Finally, datacenter customers are up against hard limits of available power and cooling capacity to fit within their budgets. It therefore has become equally essential to reduce the energy requirements of systems while also attempting to increase the performance of these given systems overall.

Accelerates Performance while reducing Energy Usage

Oracle has developed a unique Sun FlashFire Technology that provides opportunity for customers to increase performance of Oracle systems for both applications and Oracle databases. This flash technology provides customers the ability to reduce energy costs and cooling loads with increases in performance capabilities that would otherwise not be feasible. For further information on this technology please see:
http://www.oracle.com/us/products/servers-storage/storage/flash-storage/index.html

For another detailed description of the hardware of Oracle's Sun Storage F5100 array please see the following Oracle White Paper:

http://www.oracle.com/partners/en/storage-considerations-acclrtn-db-163691.pdf

I/O Performance Issues and Solutions

With database sizes used by simulation management systems growing rapidly database performance issues related to multiple concurrent users storing and retrieving large amounts of simulation related meta-data is now becoming an increasing area of focus for database administrators. In this paper we discuss experiments we have done with different storage configurations and large database configurations (including an MSC SimManager generated 235 million object, 500gb database) where we

found that providing Flash storage in place of traditional hard disk drives can provide a significant performance boost to the simulation management system's response times when accessing the Oracle database— showing between 2 - 5x speedups for typical simulation management user activities. The scaling and performance tests were done with various tools provided by MSC.Software that simulate database access and storage activities similar to those performed by an engineering organization using the simulation management system MSC.SimManager. The use case examples shown in the next section compare performance using traditional HDD (4-10x 10K rpm, SAS disks) and a SAS direct-attached Oracle Sun Storage F5100 Flash array (10-40 x 24gb Fmods).

Background

Typically two methods have been used to improve application response times and avoid the I/O bottleneck. The first method is to add more physical memory to the server so that the I/O subsystem has more capacity to "cache" I/O in high-speed physical memory. The second method is to use the following best practice configurations:

1. Striping disks for additional performance

2. Higher performance drives (SAS instead of SATA, 15K instead of 10K RPM drives)

3. Using internal or direct attach disks instead of NFS accessed storage.

What we've found in our performance studies with Flash storage devices is that Flash can provide superior performance compared to the above I/O tuning methods. For example, we have shown that with MD Nastran replacing traditional HDD devices with Flash storage can provide the same performance boost (reduced elapsed times) as doubling the size of the physical memory on a server but at a much lower cost. In other cases we have shown that by replacing the HDD devices with Flash storage we can reduce the run times of highly computational and I/O intensive applications like MD Nastran between 2-4 times and at the same time more fully utilize the available compute capacity within the latest high-performance and high-core count servers powered by Intel's new processors. In addition, simulation management codes like MSC's SimManager that rely on the Oracle database, have shown up to 5 x speedups when the database resides on flash compared to HDD devices.

SimManager Use Cases

The following three use cases illustrate different database configurations where we saw a 3 to 5x performance boost for typical MSC's SimManager user activities. The use cases measured performance by simulating the typical SimManager user activity of either searching for data stored for all projects in the database or just data specific to one specific simulation project.

Use Case Recommendation:

Of the three use cases described below the recommendation for when to apply each configuration is:

1. For users who want the maximum performance boost and have the budget to purchase the necessary Flash storage the recommendation is to use the configuration described in Use Case#1 (database is all on Flash) along with Use Case#3 (flash cache to extend the SGA space).

2. For users who only have enough Flash storage to hold the database but still want to see dramatic improvements the recommendation is use the configuration described in Use Case #1 (database is all on Flash).
3. For users who don't have enough Flash storage to hold the entire database the recommendation is use either the configuration for Use Case#2 (temporary tablespace only on Flash) or if the criteria are met as described below then Use Case#3 (flash cache to extend the SGA space).

The performance testing tools/scripts used in these examples is a script that performs SQL selects (searches) through the database, returning the total number of objects (e.g., users→projects→milestones→disciplines→results→keyResults).

Use Case#1: Multiple concurrently running database searches (a read intensive database operation that counts all database objects). The database used in this test contained 117 million objects.

This example includes four concurrently running database SQL queries that read and count objects in the database. The results of this test showed approximately a 5x speedup in the time required to complete the database operations when the database resided on Oracle's Sun Storage F5100 Flash array compared to 10k RPM SAS disks.

The following Figures [1-3] show some key I/O related statistics gathered during this benchmark that illustrate the reasons for the 5x speedup with Flash (Number of IOPS and Megabytes per second (MB/sec) being increased as well as lowering the latency of the I/O):

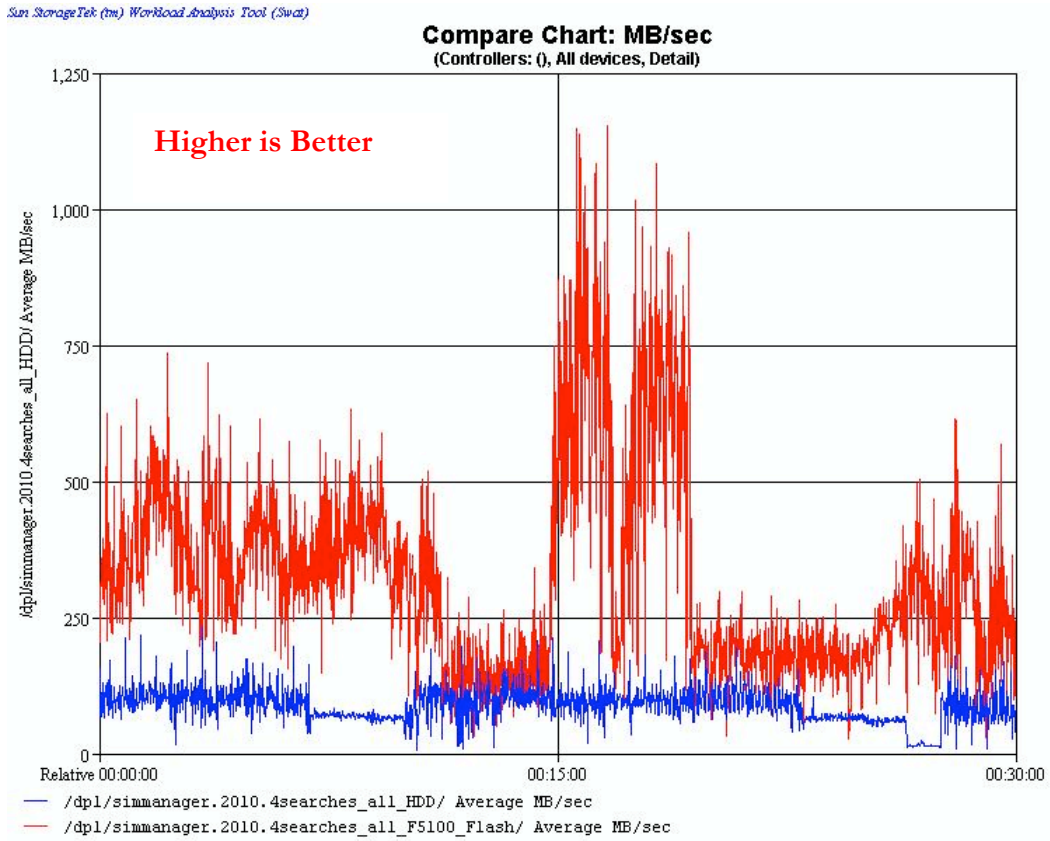


Figure 1. Flash Technology Increases Database Throughput. MB/sec comparison for Disk and Flash [Red=FLASH, Blue=HDD]

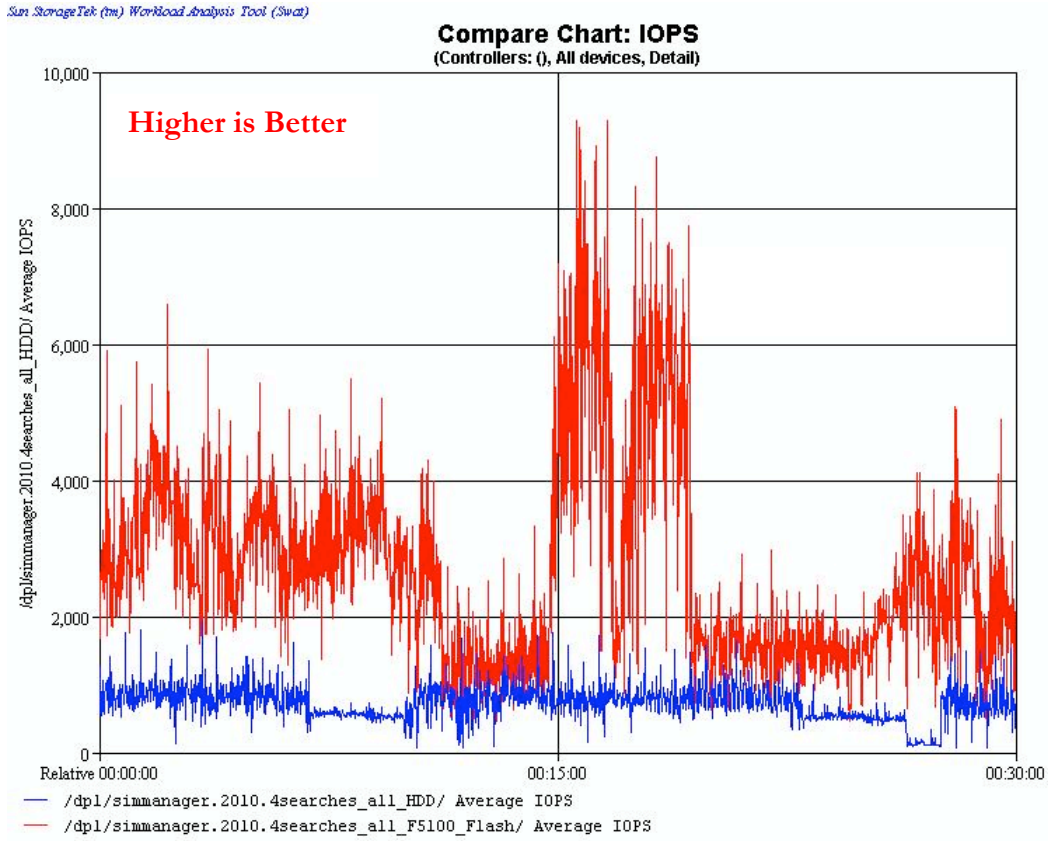


Figure 2. Flash Technology Increases Database Throughput. IOPS comparison for Disk and Flash [Red=FLASH, Blue=HDD]

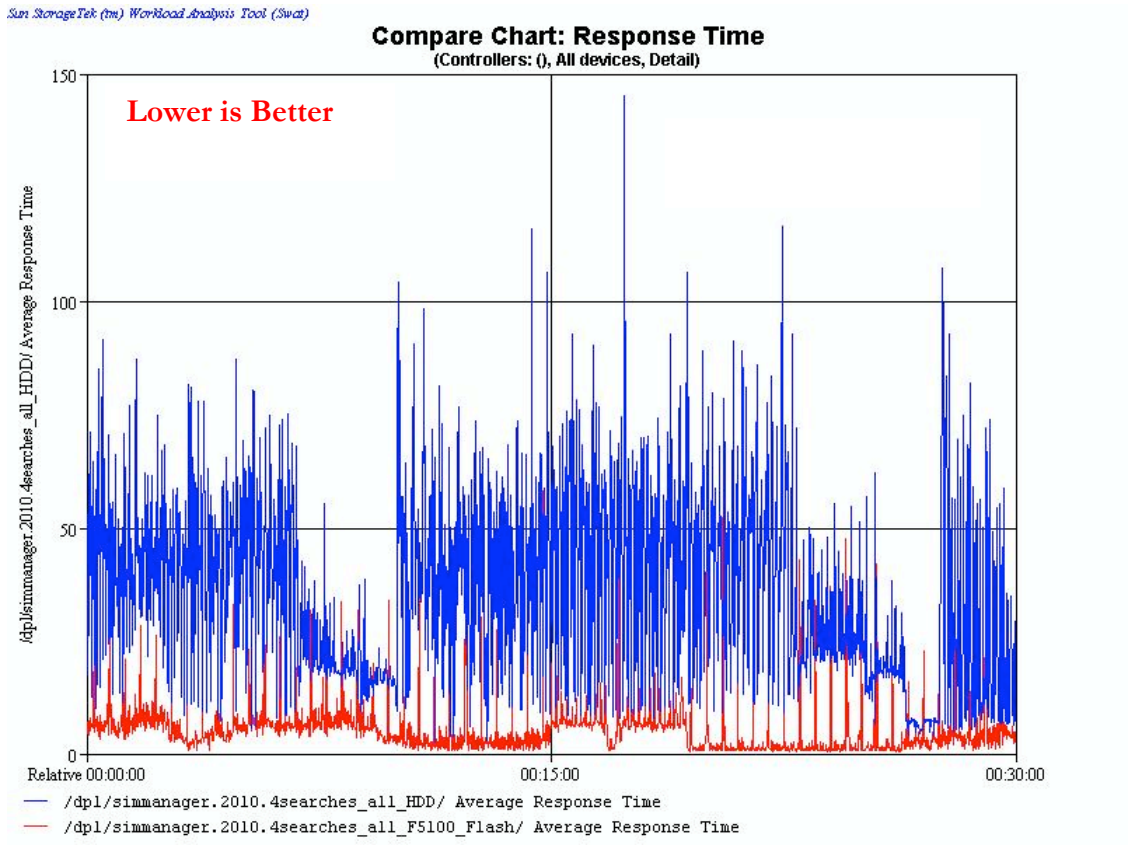


Figure 3. Flash Reduces Database Latency. Response Time comparison for Disk and Flash [Red=FLASH, Blue=HDD]

The above response time chart shows the I/Os on the hard disk drives have almost 12x more I/O latency time (i.e., 12x slower I/O response times) than when the database files reside on Flash storage. This faster response time on the Flash storage is due to the random read nature of the database access operations in this example and the fact that solid-state storage (Flash) does not have the physical disk head movement or disk platter rotation overhead of traditional hard disk drives.

Use Case #2: Speedups of 2.3x by partial use of Flash [temporary tablespace and logs]

If the Flash storage available is insufficient to store the entire database then significant speedups are still possible by selectively moving only a subset of the tablespaces and logs from HDD (hard disk) to Flash storage.

--Temporary tablespace: Moving only the temporary tablespace, comprising in this case approximately 100GB (less than 1/3 of the total database), resulted in a 2.3x speedup. This was performed with a 103 million object database.

The MB/sec [Figure 4] below illustrates this speedup. Note the following:

- All database files on HDD **[BLUE]**
- All database files on HDD except for the temporary tablespace on Flash **[RED]**

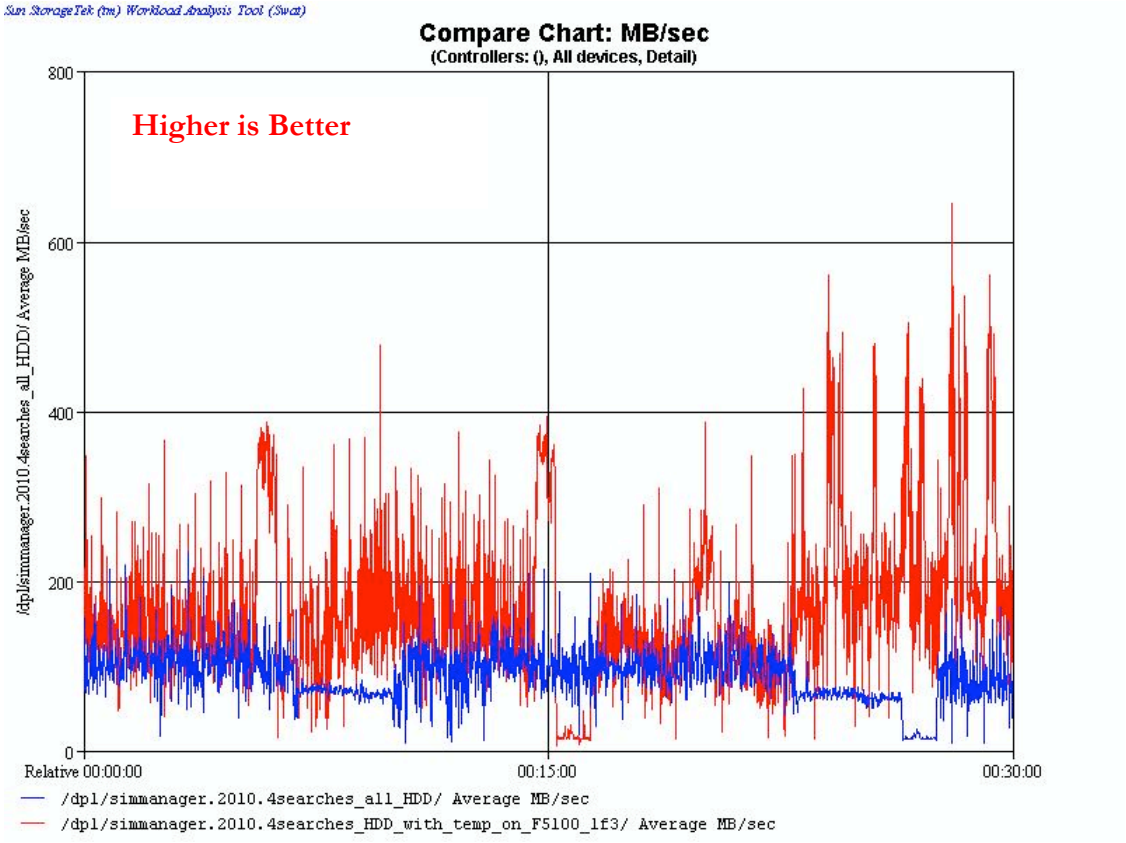


Figure 4. Flash Increases Database Throughput. MB/sec comparison for Disk and Flash [Red=FLASH, Blue=HDD]

Use Case #3: Oracle Flash Cache *(new in Oracle 11g (release 2))*

The purpose of this use case was to test Oracle's new flash cache feature to extend the Oracle 11g SGA space. In Oracle databases, the System Global Area (SGA) is a group of shared memory areas that are dedicated to Oracle "instance" (Oracle processes in execution sharing a database). All Oracle

processes use the SGA to hold information. The SGA is used to store incoming data (data and index buffers) and internal control information that is needed by the database. The size of the SGA is limited by the size of the available physical memory. The following test measured the performance of a new Oracle Database 11g (Release 2) feature which allows extension of the SGA size and caching beyond the physical memory to flash storage such as Oracle's Sun Storage F5100 array.

This study involved testing various combinations of database sizes, machine physical memory, and flash cache sizes, with the goal of understanding which customer configurations will get the most benefit from using this new feature.

Here's a summary of the findings and best practice recommendations from those tests where we were able to get a 3x performance boost with this new flash cache feature in Oracle 11gR2—what is impressive about this 3x performance boost is that the improvement was "in addition" to the performance improvements we already observed comparing Flash to HDD as described in Use Case #1.

- Use of Flash Cache can boost performance significantly. When one uses the Flash Cache feature of 11gR2 results will be dependent upon the amount of memory a system has and the amount of System Global Area (SGA) defined, the amount of flash cache being requested, and the working set size at any point in time. When choosing the size of the flash cache keep in mind that for each database block moved from the buffer cache to the flash cache, a small amount of meta-data about the block is kept in the buffer cache. For a single instance database, the meta-data consumes approximately 100 bytes. For an Oracle Real Application Clusters (Oracle RAC) database, it is closer to 200 bytes. You must therefore take this extra memory requirement into account when adding the flash cache. For further details please see "Tuning Memory for the Flash Cache" in Oracle's Database Administrator's Guide 11g Release 2 (11.2).
- Since reducing the available buffer cache can degrade overall database performance it's recommended you experiment with different sizes of flash cache to find the optimum size.
- In addition, the working set of your database operations should fit within the "flash cache". For example, we performed a SQL database search operation on the entire MSC SimManager 235 million object, 500 GB database used in our benchmarking and saw no performance improvement—the reason being that the working set overflowed the 100 GB flash cache that we assigned for this test. However, when we reduced the search operation and the flash cache to a more reasonable size working set (50 GB flash cache and a 1.2 million database object search) we saw a 3x performance improvement due to the reduced working set required by the search operation now being directly accessed from the flash cache.
- Once written, flash cache buffers are read-only and updates are only done into main memory SGA buffers. So, this feature is expected to primarily benefit read dominated workloads.

Two new init.ora parameters were used to activate the Oracle 11gr2 "Flash Cache" feature:

```
db_flash_cache_file=/lf/flash_cache

db_flash_cache_size=49G
```

The db_flash_cache_file parameter takes a single file name, which can be a filesystem, a raw device, or an ASM volume. The db_flash_cache_size parameter specifies the size of the flash cache. The flash cache can be increased/decreased or deactivated by the SQL alter command—for example:

```
SQL> show parameter flash
NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
db_flash_cache_file                  string      /lf/flash_file
db_flash_cache_size                  big integer 49G
db_flashback_retention_target        integer     1440

SQL> alter system set db_flash_cache_size=0G;
System altered

SQL> show parameter flash
NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
db_flash_cache_file                  string      /lf/flash_file
db_flash_cache_size                  big integer 0G
db_flashback_retention_target        integer     1440

SQL> alter system set db_flash_cache_size=49G;
System altered

SQL> show parameter flash
NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
db_flash_cache_file                  string      /lf/flash_file
db_flash_cache_size                  big integer 49G
db_flashback_retention_target        integer     1440
```

The Hardware/Software Configuration

The following hardware and software configuration described in Table 5 below was used for the use cases described in this white paper. MSC SimManager and the Tomcat web server were run on the Sun Fire X4170 while the database was running on the SunFire X4270 (listed below in the detailed description as Server #1 and Server #2—one server was installed with Oracle 10g and the other with Oracle 11gR2). Due to time and resource constraints we did not include the results in this paper from our M-Series tests (however, from the tests we did complete we saw similar performance boosts on the M-Series to those we reported for the Sun Fire X4270 (Server #1 and Server #2)).

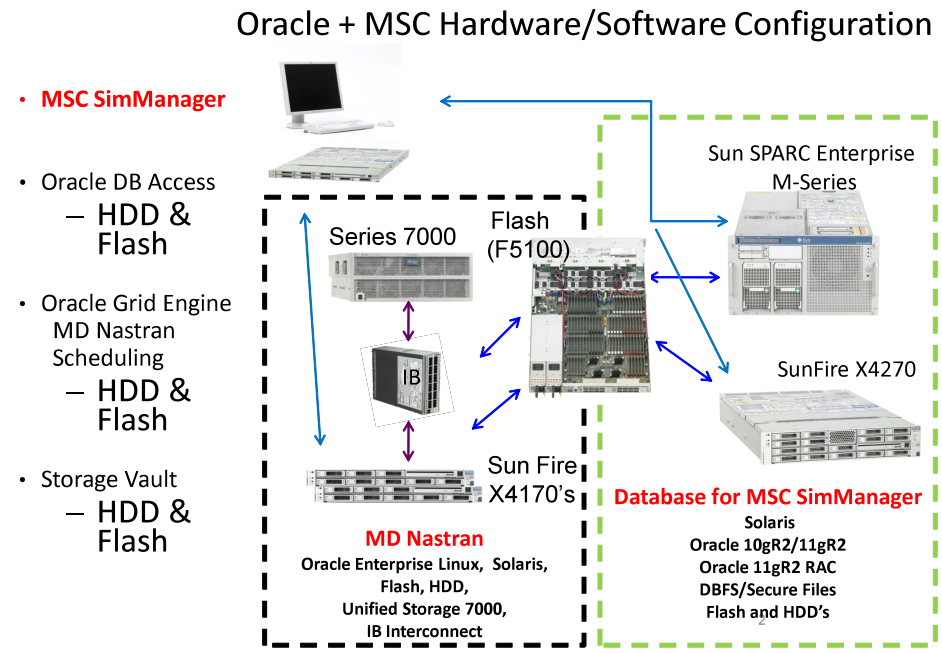


Figure 5. Hardware/Software Configuration

Web/Application Clients

- Product Name: Sun Fire X4170
- Processor: Intel 2.93GHz Quad-Core Xeon X5570 (8MB L2, 6.4 GT/s QPI, SMT, Turbo, 95W),
- Memory: DDR3 1333 MHZ, 24GB,
- Disk: 146GB 10K RPM SAS (2 for OS mirror, 4 for scratch) ,
- Flash: 6 x 24GB Flash Modules
- OS: Solaris 10

Database Servers

Server #1:

- Product Name: Sun Fire X4270
- Processor: Intel 2.93GHz Quad-Core Xeon X5570 (8MB L2, 6.4 GT/s QPI, SMT, Turbo, 95W),
- Memory: DDR3 1333 MHZ, 24GB,
- Disk: 146GB 10K RPM SAS (2 for OS mirror, 4 for scratch) ,
- Flash: 20 x 24GB Flash Modules
- OS: Solaris 10
- Database: Oracle 10g

Server #2:

- Product Name: Sun Fire X4270
- Processor: Intel 2.93GHz Quad-Core Xeon X5570
- Memory: DDR3 1333 MHZ, 24GB
- Disk: 146GB 10K RPM SAS (2 for OS mirror and 4-10 for scratch) ,
- Flash: 40 x 24GB Flash Modules
- OS: Solaris 10
- Database: Oracle 11gR2

Server #3:

- Product Name: Sun Sparc Enterprise 5000

- Processor: SPARC64 VI: 2.15GHz Dual Core processors, SPARC V9
- Memory: 8GB
- Disk: 2 x 73GB SAS Disk Drives
- Flash: 10 x 24GB Flash Modules
- OS: Solaris 10
- Database: Oracle 11gR2

Software:

- Oracle Grid Engine
- Oracle Studio 12 software
- Oracle HPC ClusterTools

Conclusion

In the introduction of this paper we described the benefits which simulation brings to the industry and the resulting challenges to effectively manage the amount of simulation data and processes. Current industry trends in the Aerospace, Automotive, Shipbuilding, electronics, consumer goods, and other industries will further increase the number and size of simulations. Simulation management systems are a recognized solution to these problems. However, the huge amount of data to be managed poses significant challenges on the underlying infrastructure, in particular with respect to database and storage technologies.

This paper has shown that introducing Oracle's Sun FlashFire technology will assist customers in meeting both their service level agreements and the protection of their initial IT investments through increased performance improvements for both compute and database services. The use cases described in this paper show that customers can potentially achieve at least 2-5x speedups on their existing Oracle Sun server infrastructure. These use cases have been run on Solaris 10 and are applicable to Oracle 10gr2 and 11gr2. [The benchmark using "Flash Cache" is specific to Oracle 11gr2.]. Oracle's Sun storage F5100 Flash array can be utilized by both Oracle's Sun database servers and Oracle's Sun computational servers based upon overall storage requirements combined.

To meet the intense I/O demands of simulation software, a storage architecture needs to include a balance of both local and network attached storage and support the ability to utilize both Flash storage and traditional hard disk drives. Following this best practice will help eliminate the I/O bottleneck as seen on traditional systems.

For the compute portion, a system that supports multiple nodes and high memory density in a compact form factor can provide an ideal base building block. Adding Oracle's Sun FlashFire technology will increase performance on these systems.

Oracle's Sun FlashFire technology added to Oracle's Sun database servers will help customers to easily meet the demands of the day and flexibly meet their demands of tomorrow without requiring a forklift to do so. Using these technologies customers can increase today's performance and are ready to scale their infrastructure for future requirements. This allows them to further expand the usage of simulation management systems, finally helping them to reach their business goals by developing more and better products and derivatives in less time.

Appendix A: Additional Database Acceleration Techniques

In additional to the use case studies discussed earlier in this paper studies have been done at Oracle showing improved database response times by placing the Redo, Undo, Archive Logs files on flash storage. These studies showed that moving these logs to flash improved database response times by over 2x. Note: We have not yet experimented with this technique using MSC SimManager.

Appendix B: Maximizing Compute Server Simulation Throughput

Background

Earlier in this paper we presented best practice recommendations for getting improved performance using Oracle's Sun Flash storage for the database access portion of the simulation management system. Over the past year we have also done numerous studies on determining "best practice" for the simulation codes that are submitted by the simulation management system. Since the simulation codes are highly compute and I/O intensive they are a critical component to be considered for a balanced system that provides optimum overall simulation throughput. The following sections describe some best practice performance recommendations that resulted from studies done with the simulation application MD Nastran.

Run more jobs concurrently

With the goal of avoiding I/O bottlenecks, the conventional recommendation has been to reduce the number of concurrent MD Nastran jobs running on any one machine. However, our studies have shown that with the recent improvements in compute processing power and large memory configurations adding Oracle's Sun Flash storage can lead to surprisingly good performance on a much larger number of concurrently running MD Nastran jobs than ever before. Here are a few examples illustrating the remarkable performance/throughput using multiple processing threads on the latest high performance and high core/thread count servers from Oracle's Sun servers powered with Oracle's Sun Flash storage.

Example #1

The following benchmark illustrates the 2.4x performance boost from using Oracle's Sun Solid State drives instead of 10k RPM HDDs. This benchmark consisted of eight analysis jobs from MSC's standard MD Nastran benchmark suite running concurrently on a single server, generating a combined 1.6TB's of I/O.

Configuration: 2 x Intel 5570 2.93GHz processors, Quad Core, 24GB RAM 1333 MHz

Total I/O: 1.6 TBs

Speedup with SSDs compared to HDDs→**58% reduction in elapsed time**

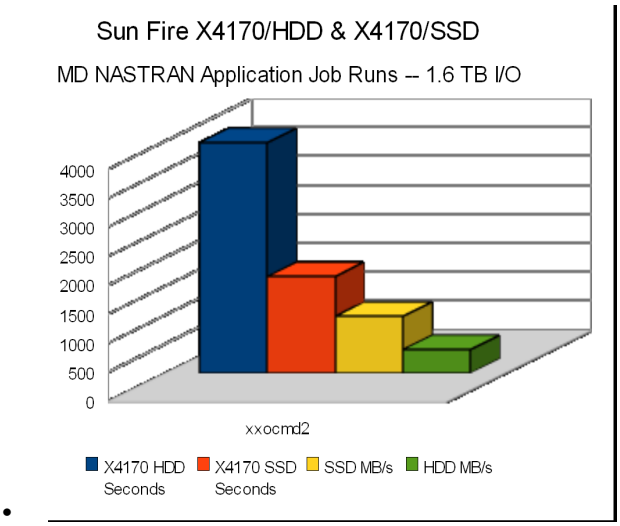


Figure 6. Benchmark showing 2.4x speedup with SSDs compared to HDDs

Example 2

The purpose of the following benchmarking was to evaluate the performance of the Westmere-EP based server and its capacity to handle the 50% higher number of computational threads available compared to prior servers.

Performance Highlights

The most noteworthy performance advantage found with the Westmere-EP based server compared to its predecessor, the Nehalem-EP, was the addition of 50% more processing threads on the Westmere-EP (a total of 24 threads). Table 6 below shows an example of the relatively small cost/overhead of taking advantage of these additional threads on the Westmere-EP server. The example shown in the table shows **approx. 20% overhead** (increased elapsed times) in order to get **50% more performance/throughput** than on the prior Nehalem-EP servers --- a reasonable cost for being able to run an additional eight MD Nastran jobs (a total of 24 jobs running concurrently compared to only 16 on the Nehalem-EP based servers)—an impressive performance improvement on this new generation of processors.

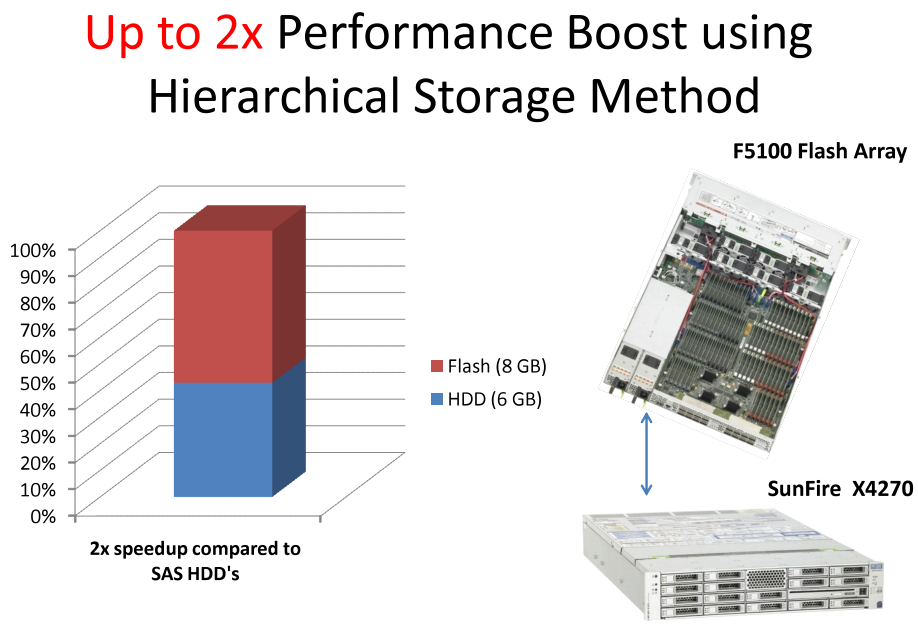
TABLE 6: WESTMERE-EP VS NEHALEM-EP PERFORMANCE (WITH ALL CORES ACTIVE ON EACH SERVER: 16 ON THE NEHALEM-EP (INTEL XEON X5570) AND 24 ON THE WESTMERE-EP (INTEL XEON X5600)). THIS TEST WAS DONE USING ORACLE'S SUN STORAGE F5100 FLASH ARRAY FOR THE SCRATCH SPACE.

MD Nastran Benchmark	CPU type	# of active cores/threads (i.e., # of Nastran jobs running concurrently)	Total Memory Used	Total I/O	Average Elapsed Time (Results in seconds)
xl0tdf1	Nehalem-EP	16	8GB	3.2TB	6094
xlotdf1	Westmere-EP	24	12.5GB	4.8TB	7423

The above data shows that the 8 additional processing threads on the Westmere-EP server (as compared to the 16 total threads on Nehalem-EP) **increased throughput by 50%** while only increasing the average elapsed time for the 24 concurrent Nastran jobs by appox. 20%. **NOTE: The use of Oracle's Sun Storage F5100 Flash array in these tests provided a 5x speedup for these 24 jobs compared to using traditional 10K RPM SAS drives.**

Example 3

Benchmarks have shown that even with only partial use of flash storage significant performance boosts are possible. The following table shows the results of a study we conducted where the scratch storage used by a series of concurrently running simulation jobs (MD Nastran) was only partially allocated to Flash [60% to Flash and the remaining 40% to HDD's]. The elapsed times for the simulation runs were reduced by 50% (a 2x performance boost) compared to using all HDD storage.



Appendix C: Scheduling Software

To efficiently use all system resources it is recommended that scheduling software be part of any simulation management solution. MSC SimManager provides a standard interface to such scheduling software so that simulations can be submitted directly from the simulation management software to the compute cluster. For detailed step-by-step instructions on how to configure Oracle's scheduling software for optimum throughput and resource utilization for simulation codes please see: http://blogs.sun.com/4HPCISVs/entry/sun_grid_engine_and_md.

Oracle is committed to developing practices and products that help protect the environment

Oracle Database Optimization for MSC SimManager using Oracle's Sun FlashFire Technology
September 2010
Author: Dale Layfield and Larry McIntosh
Contributing Authors: Stefan Mayer, Mark Norris and Milan Rosina

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

SOFTWARE. HARDWARE. COMPLETE.