Oracle White Paper
April 2010

# Oracle Database 11g Release 2 Improving OLTP System Performance by Using Database Smart Flash Cache with Fujitsu SPARC Enterprise



.

## Introduction

Since concluding an OEM agreement in 1989, Oracle Corporation Japan (“Oracle Japan”) and Fujitsu Limited (“Fujitsu”) have engaged in various joint activities, including system construction, joint validation, and post-installation support, with the overall goal of providing solutions that deliver safety and security to clients.

In November 2006, Oracle Japan established a collaborative system with Fujitsu and other grid-strategy partners, developing the advanced Oracle GRID Center[1] with the goal of building next-generation grid-based business solutions to optimize corporate system platforms.

Lending its full support for the establishment of the Oracle GRID Center, Fujitsu has been active in joint technical validation experiments with Oracle Japan involving its servers and storage products.

This paper discusses the advantages of using Database Smart Flash Cache, a new feature provided in Oracle Database 11g Release 2, in an OLTP system configured with the SPARC Enterprise M3000 and ETERNUS DX80.

[1] The Oracle GRID Center provides many technical reports.

Oracle GRID Center (http://www.oracle.co.jp/solutions/grid_center/index.html)

Overview ........ 3
Purpose of Validation ........ 4
Summary of Validation Results ........ 4
    HDD and SSD I/O Performance Measured with Oracle ORION ........ 4
    Comparison of OLTP-based Application Performance ........ 4
Description of Oracle Product Function ........ 6
    Database Smart Flash Cache ........ 6
Description of Platform ........ 7
    Fujitsu SPARC Enterprise ........ 7
    ETERNUS Storage System ........ 8
Environment for validation ........ 12
    System Configuration ........ 12
Validation Results ........ 15
    HDD and SSD I/O Performance Measured with Oracle ORION ........ 15
    Comparison of OLTP-based Application Performance ........ 20
Summary ........ 36
Appendix A ........ 37
    Tuning Database Smart Flash Cache ........ 37
Appendix B ........ 40
    Overview of SPARC Enterprise ........ 40
    ETERNUS DX series ........ 41

## Overview

The majority of corporate business operations have adopted advanced IT systems in recent years. Coupled with changing laws and compliance requirements, this is expected to increase the volume of data processed by IT systems threefold during the next two years.

Many users of IT systems believe most problems associated with growing data volumes are generated in data warehouse systems. In fact, similar problems can also occur in on-line transaction processing (OLTP) systems. A conventional OLTP system caches most of its data in its database buffer cache (hereafter referred to as a “buffer cache”), thus achieving maximum use of CPU resources and high-speed processing. However, increased data volumes reduce buffer cache hit rates, resulting in frequent input and output of data to and from slow-speed hard disk drives (HDDs). This impedes the effective use of CPU resources in many cases and lowers OLTP system performance.

These problems are generally resolved by adding memory chips to database servers to increase the size of the buffer cache. However, memory chips are expensive, and restrictions limit the amount of memory that can be mounted on servers. Thus, we must increase the speed of input and output to and from storage devices to handle growing I/O rates. One approach is to increase the number of HDDs for I/O dispersion (striping). However, to gain high storage I/O performance in balance with CPU performance requires vast numbers of HDDs. Since this approach results in increased power consumption and larger installation space, it is less acceptable due to the need to reduce IT costs (see Fig. 1).

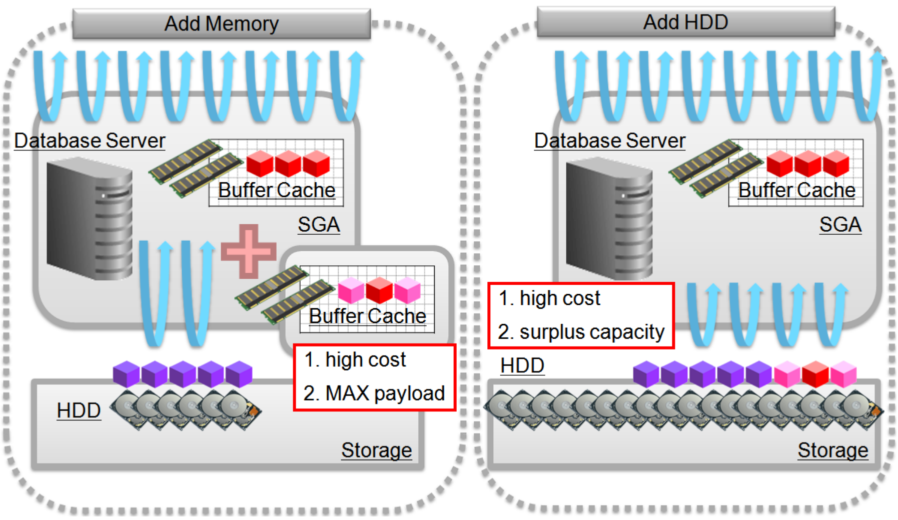



**Fig. 1 Resolving bottlenecks in storage I/O performance**

Oracle Database 11g Release 2 features Database Smart Flash Cache[2], which makes it possible to deploy solid state drives (SSDs) to resolve the above-mentioned problem.

This paper discusses the use of Database Smart Flash Cache in an OLTP system configured with the SPARC Enterprise M3000 and ETERNUS DX80 for resolving the bottleneck in storage I/O performance and improving the OLTP system performance at low cost.

## Purpose of Validation

There are two methods for using SSDs in a database system:

- To replace HDDs
- Database Smart Flash Cache

We performed validation tests at the Oracle GRID Center to confirm the advantages of each method of SSD usage and to establish best practices for using SSDs in database systems.

## Summary of Validation Results

The details and results of the validation experiments performed at the Oracle GRID Center are summarized below. For detailed information on each item, please refer to the corresponding chapter in this report.

### HDD and SSD I/O Performance Measured with Oracle ORION

We used Oracle ORION, a tool used to simulate I/O workloads on an Oracle database, to measure HDD and SSD I/O performance. I/O performance measurements confirmed the high IOPS performance of the SSD, particularly for random reads, compared to the HDD. Measurements also confirmed that in the case of random I/O mixed with reads and writes, the maximum IOPS value of the SSD declined as the percentage of writes increased.

Details ⇒ HDD and SSD I/O Performance Measured with Oracle ORION

### Comparison of OLTP-based Application Performance

#### Performance of system deploying SSDs in place of HDDs

[2] Database Smart Flash Cache is supported by the Enterprise Edition and by Sun SPARC Solaris, Sun x86-64 Solaris, and Oracle Enterprise Linux platforms.

We created operating environments in which a database schema was configured on HDDs and on SSDs and compared OLTP processing performance. The measurements confirmed that configuring the data schema on SSDs removed bottlenecks to IOPS performance found in systems with HDDs, while also boosting OLTP system performance.

Details ⇒ Configuration of database schema on HDDs and on SSDs

With large-scale databases, it is impractical to configure the entire database schema on SSDs. In such cases, the user can analyze and identify the schema objects read frequently from the HDDs, then deploy them on SSDs before other objects. We created such an environment and measured the OLTP processing performance. The results confirmed improvements in OLTP system performance, but also indicated SSDs were not used to full advantage.

Details ⇒ Selective configuration of schema objects on SSDs

### Performance achieved with Database Smart Flash Cache

The Database Smart Flash Cache feature provided in Oracle Database 11g Release 2 deploys SSDs as a database cache, and automatically caches data blocks that are not retained in the buffer cache in the Database Smart Flash Cache area. Without performing analysis or operation similar to those performed to assess the performance of the system deploying SSDs in place of HDDs, we confirmed that this improved OLTP system performance by enabling maximum use of SSDs.

Details ⇒ Performance achieved with Database Smart Flash Cache

# Description of Oracle Product Function

## Database Smart Flash Cache

Oracle Database retains data blocks obtained from HDDs inside the buffer cache in main memory. This makes it possible to retrieve data blocks cached in the buffer cache without generating I/Os to or from HDDs for faster response. Additionally, the Least Recently Used (LRU) algorithm automatically manages cached data blocks so that those with high access frequency are given priority status for retention in the buffer.

SSDs offer significantly higher IOPS performance than HDDs and have attracted significant attention in recent years. To use SSDs, which cost more than HDDs but less than memory chips, we developed a new approach wherein database files are stored on SSDs rather than HDDs. However, as of April 2010, the relatively high cost per unit of capacity of SSDs makes it impractical to use SSDs to store data whose volume can be expected to continue growing.

Database Smart Flash Cache uses SSDs as an extended area of the buffer cache (Level 1 cache) to provide a Level 2 cache for data blocks (see Fig. 2). In the Database Smart Flash Cache area, data blocks which are not retained in the buffer cache are automatically arranged by an algorithm with a performance superior to the LRU algorithm. This allows high-speed access to data blocks that were not previously retained in the cache due to the limitations on the size of the buffer cache, despite their high access frequency. As a result, Database Smart Flash Cache can improve OLTP process performance at costs much lower than adding memory chips of an equivalent capacity.

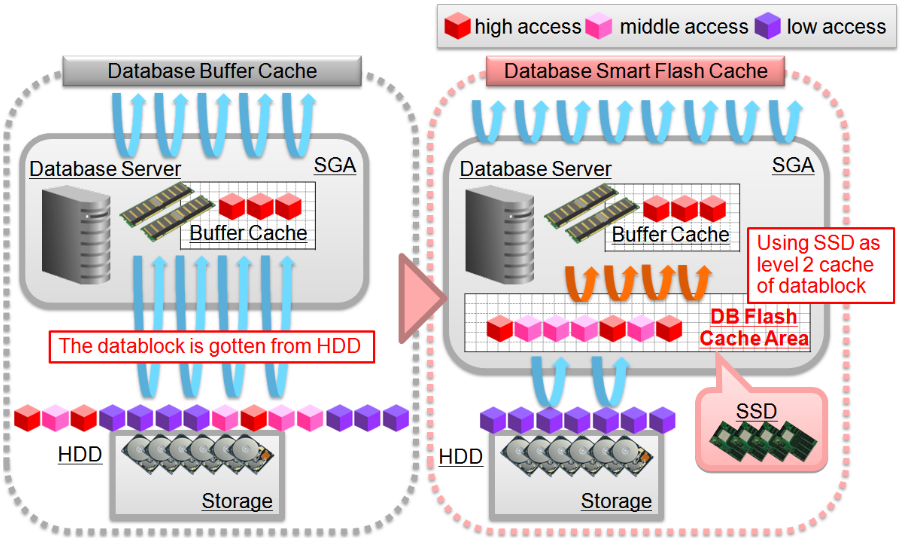



**Fig. 2 Application of Database Smart Flash Cache**

Only data blocks written to the HDDs (clean blocks) are assigned to the Database Smart Flash Cache area. This allows the user to back up data files in the same way as before, without needing to track what data is on the SSDs. Moreover, since only clean blocks are cached, data integrity is maintained even if a problem occurs in the SSDs.

# Description of Platform

## Fujitsu SPARC Enterprise

The roles of IT systems in today's businesses continue to grow ever more important. As the data volumes handled by companies have grown, companies have begun operating servers for a wide range of applications, such as Web servers, application servers, and database servers. Meanwhile, companies face several emerging issues, including inadequate server performance leading to bottlenecks in IT systems, increased operating costs due to growing numbers of server units, and the increasing weight of corporate social responsibilities and the growing role of IT systems in protecting the global environment. Resolving these problems will require cost reductions, improved efficiency of IT systems, and curtailment of carbon dioxide emissions through total optimization of IT systems based on physical and virtual integration of servers and middleware deployment.

The SPARC Enterprise series addresses the diverse management issues confronting today's companies through timely management, business continuity, lower TCO, and environmental performance. The SPARC Enterprise lineup includes the SPARC Enterprise M9000, M8000, M5000, M4000, and M3000, which feature reliability equivalent to that of a mainframe, making them suitable for mission-critical business applications, and the SPARC Enterprise T5440, T5240, T5220, T5140, and T5120 which offer the high throughput performance required by Web front-end applications.

### SPARC Enterprise M3000

The SPARC Enterprise M3000 incorporates a SPARC64 VII to provide multi-core, multi-threaded configurations with up to four cores and eight threads. Systems can address up to 64 GB of memory, providing the functionality required for business operations with a 2U (2-unit) space. Standard configurations include one SAS port and four PCI Express slots. Offering class-leading performance for entry-level models, the SPARC Enterprise M3000 is ideal for a wide range of applications, including application servers and database servers.

The SPARC Enterprise M3000 also features the high reliability associated with higher-end models, ensuring reliability through systems reliability at the LSI, unit, and system levels.

The SPARC Enterprise M3000 is also a environmentally-friendly product, that are qualified as Super Green Product by Fujitsu. With its 2U (2-unit) dimensions, it offers low weight and space savings of 50% over the PRIMEPOWER450 (4U). Its maximum power consumption of 505 W (at 100 V) represents a 54% reduction. Combined with performance gains, these improvements cut carbon dioxide emissions a year by approximately 65%. The SPARC Enterprise M3000 also boasts a low-noise design with operating noise levels of 47 dB at the standard server site ambient temperature of 25°C, making it the leading environmentally-friendly server for power consumption and noise levels among competing four-core servers.

The standard Solaris 10 configuration includes the virtual technology known as Solaris Container[3], which allows resources to be centralized through server consolidation, even with the SPARC Enterprise M3000, for improved system efficiency.

- SPARC Enterprise M3000 features
  - Offers leading entry-level processor performance for SPARC/Solaris.
  - Offers mid-range-class high reliability technology in an entry-level product.
  - Offers Green Policy Innovation features (energy and space savings).

## ETERNUS Storage System

Growing integration between IT and corporate management has made it essential to deploy an IT infrastructure enabling efficient use of management and other information, and supporting corporate management and social responsibility regarding risk management.

Fujitsu provides storage systems that meet wide-ranging customer requirements. The ETERNUS storage system is developed to meet the following three requirements:

1. Ensure business continuity and scalability for large data volumes without delays.
2. Ensure data integrity and security for correct data storage.
3. Ensure appropriate and flexible maintenance of enterprise-level and large-volume data while minimizing TCO.

Capable of responding to changes in operating configurations to achieve customer business goals, the ETERNUS storage system provides storage infrastructure and services that ensure access to required data and storage resources at any time from business applications and processes assigned suitable authorization levels.

### ETERNUS DX80

The ETERNUS DX80 is an entry-level disk array offering an environmentally-friendly energy-saving design, the same functions offered by higher-end models, and high reliability.

[3] Oracle Real Application Clusters does not support operations on Solaris Container.

Incorporating Fujitsu's accumulated know-how, the compact ETERNUS DX80 provides optimal storage solutions for small- and mid-sized systems demanding high cost effectiveness.

- Environmentally-friendly features
  - 8% reduction in power consumption to reduce carbon dioxide emissions a year by up to 100 kg [Note 1]
  - Optimized fan speed control for achieving operating noise levels of 6 dB (max.) [Note 1]
  - Supports low power consumption, high reliability, high performance SSDs
  - Eliminates battery to reduce industrial waste
  - Eco mode incorporating MAID technology to reduce power consumption
- Flexible operations management
  - Advanced copy function for instantaneous data replication
  - Intuitive GUI for easy setup
  - Supports lightweight 2.5-inch drives, which reduce power consumption of 3.5-inch disk drives by roughly 50% [Note 2]
- High reliability and data protection
  - Features redundant design for main components and supports RAID5+0 which offers higher reliability and performance than RAID5, and RAID6 which ensures data protection even if two disks fail simultaneously.
  - Uses Fujitsu's original system for encoding data written to disks.
  - Adds check code to all stored data to ensure data consistency.

[Note 1] Based on the results of our comparison test with a previous model (ETERNUS2000) mounting 12 450-GB (15,000 rpm) 3.5-inch disk drives at an ambient operating temperature of 25°C

[Note 2] Based on the results of our comparison test with a previous model mounting 120 300-GB (10,000 rpm) 2.5-inch disk drives and 120 600-GB (15,000 rpm) 3.5-inch disk drives (equivalent to DX80 maximum configuration)

### SSD (Solid State Drive)

The SSD is a memory device that uses flash memory. Since SSDs offer high speed and low power consumption, they are suitable for use with systems that require extremely fast processing speeds, such as large-scale database systems.

The ETERNUS DX80 can mount up to nine SSDs.

**Table 1 Comparison of HDD and SSD in ETERNUS DX80/DX90**

| | SSD | HDD |
|---|---|---|
| Speed [Note 1] | Very Good<br>Read: 16 times faster than HDD<br>Write: 11 times faster than HDD | Good |
| Power consumption [Note 2] | Very Good<br>Idling: 74% of HDD<br>I/O operation: 49% of HDD | Poor |
| Impact resistance | Good<br>Absence of moving parts means high resistance to impact and shock. | Poor<br>Susceptible to impact and shock |
| Capacity | Good<br>200 GB, 100 GB | Very Good<br>3.5" SAS disk: 600 GB, 450 GB, 300 GB<br>3.5" nearline SAS disk: 2 TB, 1 TB, 750 GB<br>2.5" SAS disk: 300 GB, 146 GB |
| Cost | Poor<br>High cost per unit of capacity | Good |

[Note 1] Comparison using ETERNUS DX80/DX90

[Note 2] Comparison of SSD unit performance and 3.5" SAS disk unit performance

The ETERNUS DX series is equipped with STEC SSDs.

With configurations deploying SSDs with the ETERNUS DX80, it is possible to configure one of the RAID groups shown in Table 2.[4]

**Table 2 RAID configurations with ETERNUS DX80 and SSDs**

| RAID LEVEL | NO. OF SSDS | CONFIGURATION NAME |
|---|---|---|
| RAID1 | 2 | RAID1 (1+1) |
| RAID5 | 4 | RAID5 (3+1) |

[4] RAID1+0 (2+2) configuration is supported from January 2010.

### ETERNUS SF Storage Cruiser and ETERNUS SF AdvancedCopy Manager

ETERNUS SF Storage Cruiser is software that simplifies setup of Disk storage systems and the management of ETERNUS storage environments. It manages operations by linking resources such as business server file systems, connection paths, mirror disks, and databases from ETERNUS disk array disk drives. ETERNUS SF Storage Cruiser allows ready identification of the correlation between individual resources and enables storage system expansion, fault recovery, and accurate information acquisition and display.

In combination with the ETERNUS disk array, ETERNUS SF AdvancedCopy Manager provides high-speed backup/restore and replication operations based on the advanced copy function. The advanced copy function rapidly copies working volumes at set times to a different volume (copy volume) within the same disk array. The copy volume can then be used to perform a backup to a tape device. ETERNUS SF AdvancedCopy Manager tape server options can be used to make easy backups from disks to tape, eliminating complex procedures such as copy completion and tape backup start scheduling and management of multiple disks and tapes. The copy volume is separated from the working volume so that no overwriting occurs even if these steps are performed while actual work proceeds.

## Environment for validation

### System Configuration

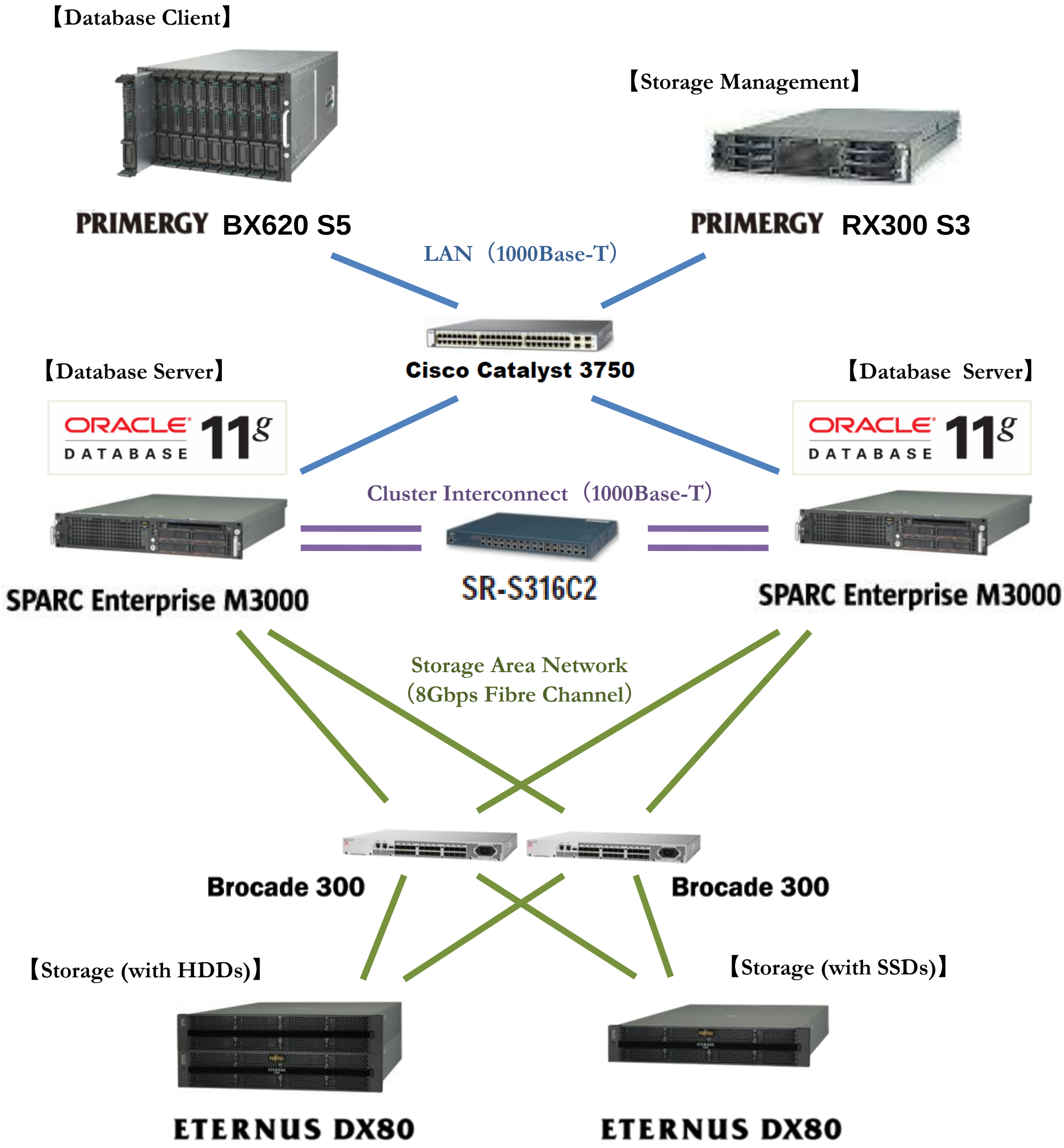



**Fig. 3 System configuration**

### Database server

With two Fujitsu SPARC Enterprise M3000 units, a 2-node Oracle Real Application Clusters was configured using Oracle Database 11g Release 2 Enterprise Edition.

| | |
|---|---|
| MODEL | Fujitsu SPARC Enterprise M3000 |
| CPU | SPARC 64 VII 2.75GHz (4 cores, 8 threads) |
| MEMORY | 32 GB |
| OS | Solaris 10 Operating System (Generic_141444-09) |
| DATABASE | Oracle Database 11g Release 2 Enterprise Edition<br>Oracle Real Application Clusters |

### Storage

Two ETERNUS DX80 units, one mounting SSDs and the other HDDs, were used.

- ETERNUS DX80 with SSDs

| | |
|---|---|
| MODEL | Fujitsu ETERNUS DX80 |
| DISK DRIVE | SSD: 100 GB x 8<br>HDD: 300 GB (15,000 rpm) SAS disk drive x 2 |

- ETERNUS DX80 with HDDs

| | |
|---|---|
| MODEL | Fujitsu ETERNUS DX80 |
| DISK DRIVE | HDD: 300 GB (15,000 rpm) SAS disk drive x 24<br>(Extended drive enclosure x 1) |

- RAID and ASM disk group configurations

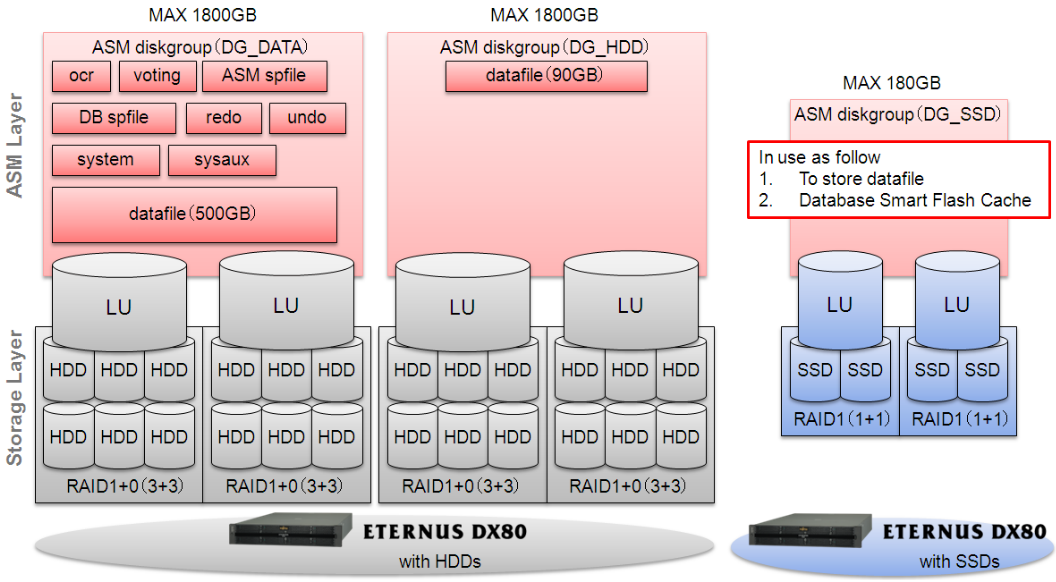



**Fig. 4 RAID and ASM disk group configuration 1 (SSD RAID1 configuration)**

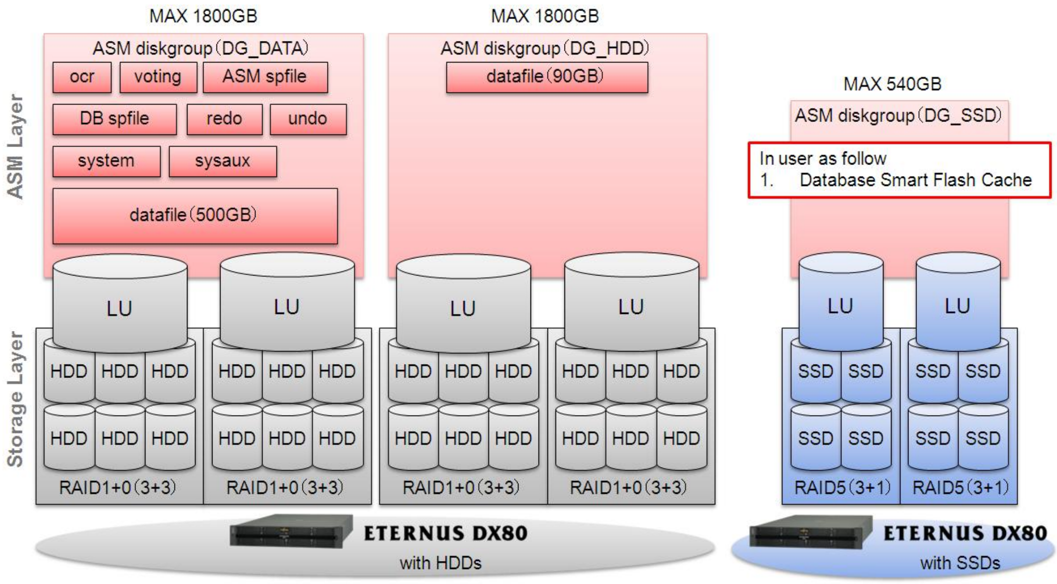



**Fig. 5 RAID and ASM disk group configuration 2 (SSD RAID5 configuration)**

### Database client

As a client, two BX620 S5 blades in the Fujitsu PRIMERGY BX600 Blade Server series were used to run business applications.

| | |
|---|---|
| **MODEL** | Fujitsu PRIMERGY BX620 S5 |
| **CPU** | Intel Xeon processor X5570 2.93GHz (4 cores) x 2 |
| **MEMORY** | 24 GB |
| **OS** | Red Hat Enterprise Linux Server Release 5.3 (Tikanga) |

### Storage management client

ETERNUS SF Storage Cruiser was used to manage the ETERNUS devices.

| | |
|---|---|
| **MODEL** | Fujitsu PRIMERGY RX300 S3 |
| **CPU** | Intel Xeon 5160 processor 3.00GHz (2 cores) |
| **MEMORY** | 3 GB |
| **OS** | Microsoft Windows Server 2003 R2 |
| **STORAGE MANAGEMENT** | ETERNUS SF Storage Cruiser 14.1<br>ETERNUS SF AdvancedCopy Manager 14.1 |

# Validation Results

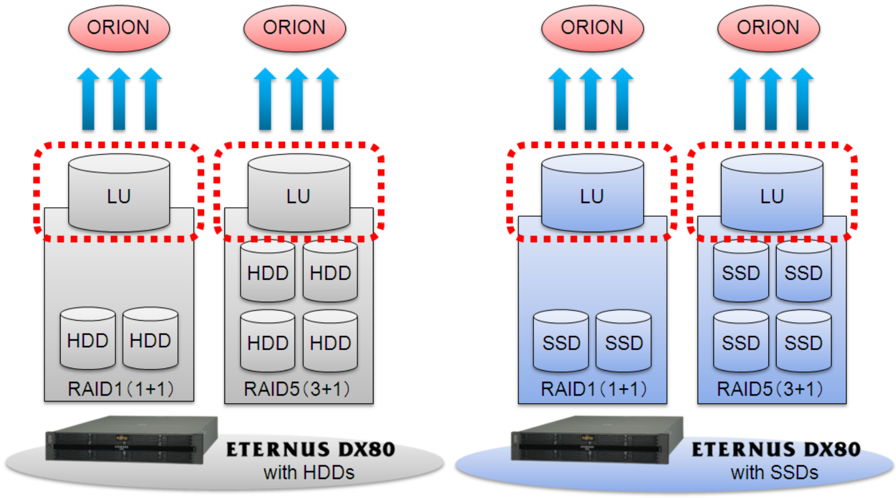
## HDD and SSD I/O Performance Measured with Oracle ORION

Before assessing the performance of the Database Smart Flash Cache, we compared the I/O performance tendency between HDDs and SSDs. For this examination, we used Oracle ORION[5] to measure I/O performance (IOPS) for random-reads and random-writes of SQL commands that are assumed to be issued from an OLTP-base application to a logical unit (LU) configured by drives. For the measurements, the write cache of the ETERNUS DX80 controller was enabled for both HDDs and SSDs.

### Random read performance

Using the RAID level and the number of drives configurable with the ETERNUS DX80 mounting SSDs, we configured a corresponding configuration for HDD - RAID group with the ETERNUS DX80 mounting HDDs, and created one LU from each RAID group. Fig. 6 shows the RAID levels and the number of drives used in the validation experiment.

[5] Oracle ORION is an I/O performance measuring tool that simulates I/O workloads on an Oracle database. Without installation of an Oracle database, it generates sequential I/O and random I/O based on assumed RAID configuration, ASM disk group, and database block size to enable the measurement of storage I/O performance. Oracle ORION is provided for different platforms and can be downloaded from Oracle Technology Network (http://www.oracle.com/technology/software/tech/orion/index.html).




| TYPE | RAID LEVEL | NO. OF DRIVES | CONFIGURATION NAME |
|---|---|---|---|
| HDD | RAID1 | 2 | HDD - RAID1 (1+1) |
| HDD | RAID5 | 4 | HDD - RAID5 (3+1) |
| SSD | RAID1 | 2 | SSD - RAID1 (1+1) |
| SSD | RAID5 | 4 | SSD - RAID5 (3+1) |

**Fig. 6 HDD and SSD storage configurations**

The following graph (Fig. 7) shows the results of measurements of random read performance (IOPS), with the I/O size set to the database block size (2 KB to 32 KB) that can be used in Oracle databases.

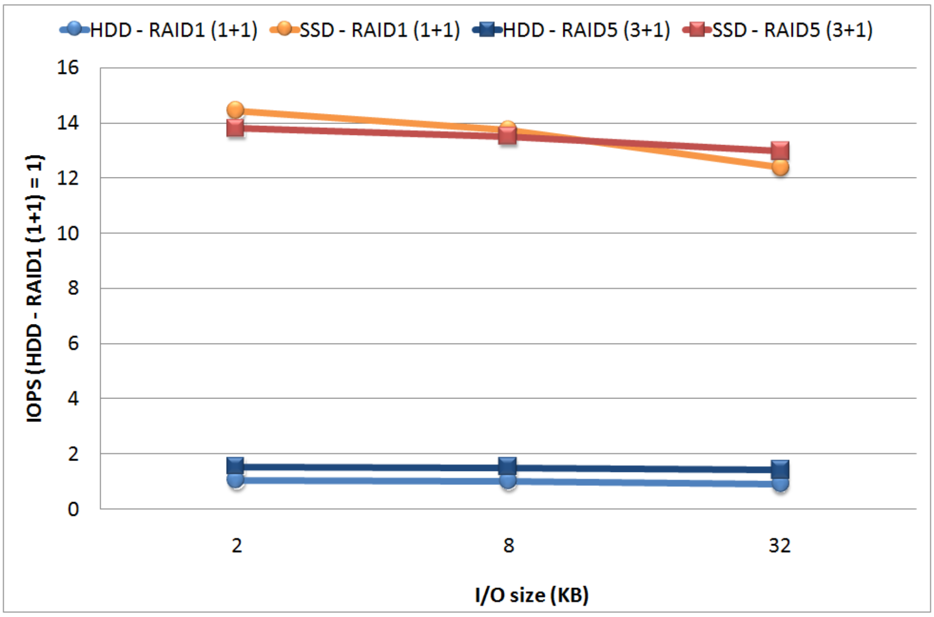
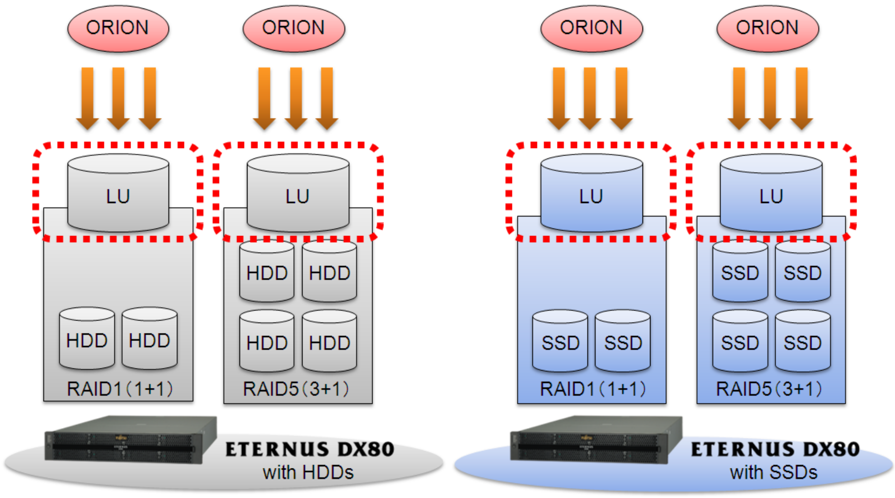



**Fig. 7 IOPS of HDD and SSD**

Based on measurements of random-read performance, we confirmed that the IOPS of the SSD exceeded that of the HDD for all I/O sizes. With the standard database block size of OLTP systems, which is 8 KB, the IOPS of the SSD was about 14 times better than that of the HDD in RAID1 and about nine times better in RAID5.

### Random write performance

For the measurement of random-write performance, we configured RAID groups similar to those used for the measurement of random read performance. Fig. 8 shows the RAID levels and the number of drives used in the validation experiment.




| TYPE | RAID LEVEL | NO. OF DRIVES | CONFIGURATION NAME |
|---|---|---|---|
| HDD | RAID1 | 2 | HDD - RAID1 (1+1) |
| HDD | RAID5 | 4 | HDD - RAID5 (3+1) |
| SSD | RAID1 | 2 | SSD - RAID1 (1+1) |
| SSD | RAID5 | 4 | SSD - RAID5 (3+1) |

**Fig. 8 HDD and SSD storage configurations**

The following graph (Fig. 9) shows the results of measurements of random write performance (IOPS), with the I/O size set to the database block size (2 KB to 32 KB) that can be used in Oracle databases.

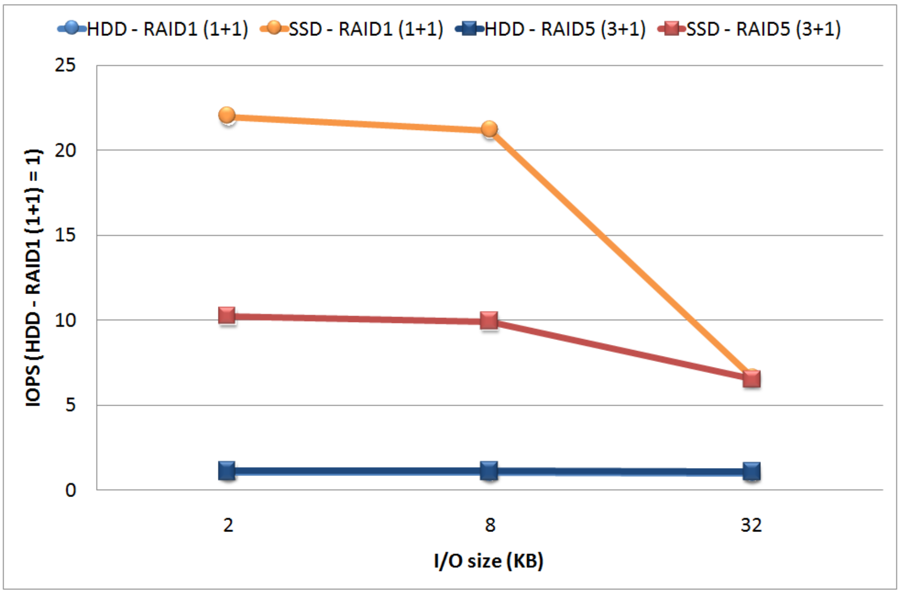



**Fig. 9 IOPS of HDD and SSD**

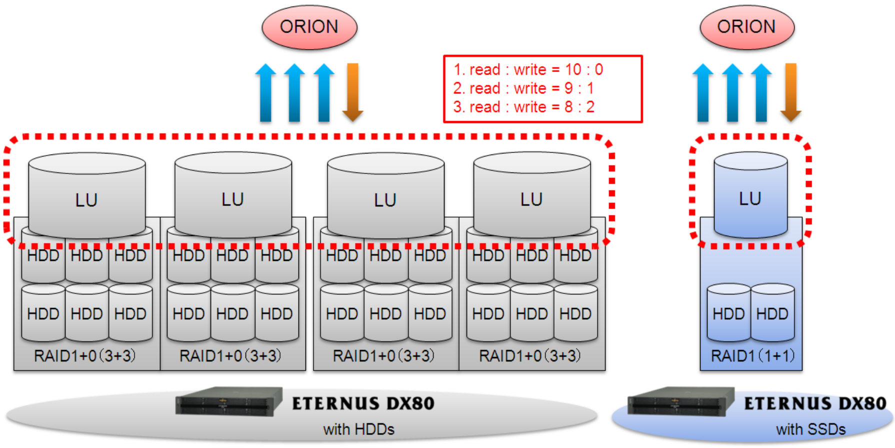
Based on measurements of random-write performance, we confirmed that the IOPS of the SSD exceeded that of the HDD for all I/O sizes. With the standard database block size of OLTP systems, which is 8 KB, the IOPS of the SSD was about 21 times better than that of the HDD in RAID1 and about nine times better in RAID5.

### Random I/O performance in mixture of reads and writes

Since OLTP systems issue SQL command for not only SELECT but also INSERT, UPDATE, and DELETE, I/O workload containing both reads and writes was generated, and IOPS measurement was performed with the I/O unit of 8 KB.

Fig. 10 shows the storage configuration used in the measurement. In the experiment described in this section, I/O was dispersed (striped) to all mounted HDDs to check the HDD configuration that could achieve the same IOPS as the SSDs in the RAID1 (1+1) configuration. Consequently, 24 HDDs were used to achieve equivalent IOPS with a configuration with two SSDs.[6]

[6] In the section, "Random-read performance," the IOPS of the SSD recorded a performance figure roughly 14 times better than the HDD, with a standard 8 KB database block size for OLTP systems. Theoretically, 28 HDDs are required to obtain the equivalent random read-performance as the SSDs in the RAID1 (1+1) configuration, but only 24 HDDs were used in the validation test due to the restriction on the number of mountable drives.




| TYPE | RAID LEVEL AND NO. OF DRIVES | NO. OF RAID GROUPS | TOTAL NO. OF DRIVES | CONFIGURATION NAME |
|---|---|---|---|---|
| HDD | RAID1+0 (3+3) | 4 groups | HDD x 24 | HDD - RAID1+0 (3+3) x 4 |
| SSD | RAID1 (1+1) | 1 group | SSD x 2 | SSD - RAID1 (1+1) x 1 |

**Fig. 10 HDD and SSD storage configurations**

The following graph (Fig. 11) shows the results of measurements of random I/O performance (IOPS), In this measurement, reads and writes were mixed and their ratio was varied. In all patterns, the IOPS of the SSD was higher. Additionally, the results confirmed that the tendency of the IOPS of the both types of drives to decrease when the percentage of writes increased.

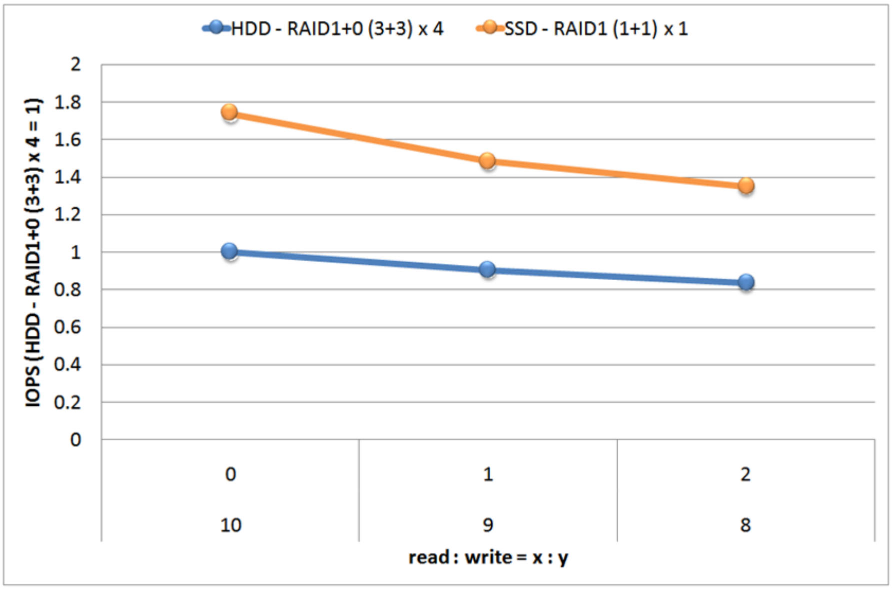



**Fig. 11 IOPS of HDD and SSD**

### Summary on the random I/O performance of HDD and SSD

Table 3 summarizes the advantages of the SSD that were confirmed based on the measurement of I/O performance using Oracle ORION.

**Table 3 I/O performance of HDD and SSD**

| DRIVE | RANDOM | |
|---|---|---|
| | READ | WRITE |
| SSD | Significantly faster than HDD | Significantly faster than HDD |
| HDD | Significantly slower than SSD | Significantly slower than SSD |

The results of experiments confirmed that the SSD offered significantly higher random I/O (IOPS) performance than the HDD. Because the SSD uses flash memory as storage media, it does not generate a seek time that the HDD needs for moving the head across a platter to access data or a search time that is required by the HDD for rotating the platter and bringing the target data to the position of the head. Therefore, the SSD offers superior random I/O performance.

## Comparison of OLTP-based Application Performance

According to measurements of HDD and SSD I/O performance using Oracle ORION, performance can be improved by using SSDs with OLTP-based applications that issue SQL and frequently generate random-I/O. In the following section, the results of validation using an OLTP-based application are described.

### Load generating tool that implements general OLTP processes

The load generating tool used in the validation experiments was a Java custom application that used Oracle JDBC OCI Driver 11g to connect to an Oracle instance and execute SQL. The sample database schema for the JPetStore application provided with Spring Framework was used to execute general SQL according to the user scenario shown in Fig. 12. The transactions used in the validation experiments are described below.

- Tx1: Transaction for searching for a product and making a purchase
  - Ordering process issues INSERT and UPDATE SQL.
- Tx2: Transaction for searching for a product only, without making a purchase
  - Only SELECT SQL is issued.

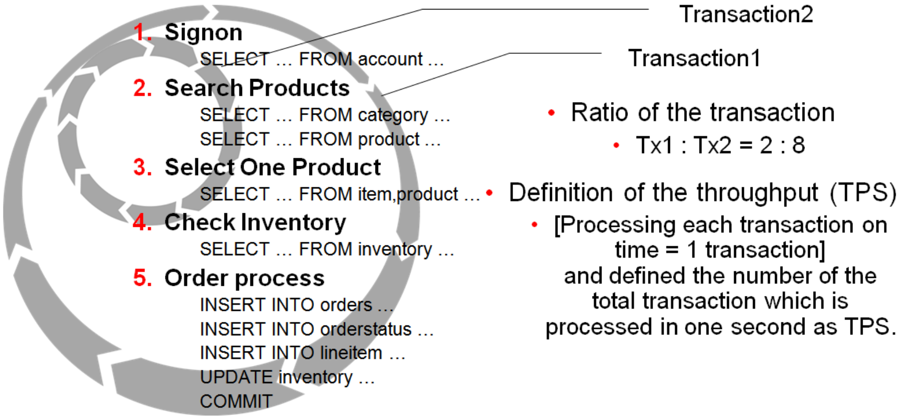



**Fig. 12 User scenario and definition of transactions**

Based on the assumption that most people use on-line shopping sites mainly for product searches, the ratio of Tx1 to Tx2 was set to 2 to 8.

In the validation, the number of simultaneous application executions (hereafter referred to as “threads”) and the range of accessed data (hereafter referred to as a “search range”) were increased to simulate an increase in the volume of data accessed by the database client in an OLTP system.

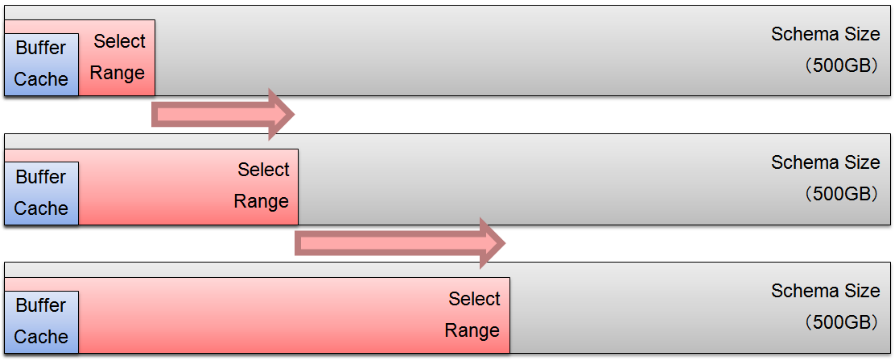



**Fig. 13 Change of search range**

### Database schema configuration

For the validation test, we prepared two types of database schemas (hereafter referred to as “schemas”), one large and one small, according to the validation items. The first schema was about 90 GB to match the maximum capacity of one LU configured with SSDs. The second schema was 500 GB to mimic a database with an increased amount of data. Details of these schema configurations are as follows.

**Table 4 Tables**

| TYPE | TABLE NAME | PARTITION | REMARKS |
|---|---|---|---|
| master | ACCOUNT | None | User |
| | PROFILE | None | User profile |
| | SIGNON | None | Password management |
| | CATEGORY | None | Product link |
| | PRODUCT | None | Product master |
| | INVENTORY | None | Product item inventory control |
| | ITEM | None | Product item management |
| transaction | ORDERS | hash (256) | Order ID, customer |
| | ORDERSTATUS | hash (256) | Date of order, condition of order |
| | LINEITEM | None | Number of orders, unit-price per order |

**Table 5 Indexes**

| TABLE NAME | INDEX NAME | REMARKS |
|---|---|---|
| ACCOUNT | PK_ACCOUNT | Main key index |
| PROFILE | PK_PROFILE | Main key index |
| SIGNON | PK_SIGNON | Main key index |
| CATEGORY | PK_CATEGORY | Main key index |
| PRODUCT | PK_PRODUCT | Main key index |
| | PRODUCT_NAME | Product name (for product search) |
| | PRODUCT_CATEGORY | Product category (for product search) |
| | PRODUCT_DESCN | Product description (for product search) |
| INVENTORY | PK_INVENTORY | Main key index |
| ITEM | ITEMPROD | PRODUCTID array |
| | PK_ITEM | Main key index |
| ORDERS | PK_ORDERS | Main key index (local) |
| ORDERSTATUS | PK_ORDERSTATUS | Main key index (local) |
| LINEITEM | PK_LINEITEM | Main key index (local) |

## Performance of system deploying SSDs in place of HDDs

What are described below are the performance improvements and advantages achieved by replacing HDDs with SSDs in an OLTP system.

### Configuration of database schema on HDDs and on SSDs

Fig. 14 shows storage configurations in which a schema was configured on HDDs and on SSDs. Tablespaces are produced in the ASM disk group (DG_HDD) consisting only of HDDs and in the ASM disk group

(DG_SSD) consisting only of SSDs, and a schema of approximately 90 GB, which is the maximum capacity of DG_SSD, was placed in each tablespace.

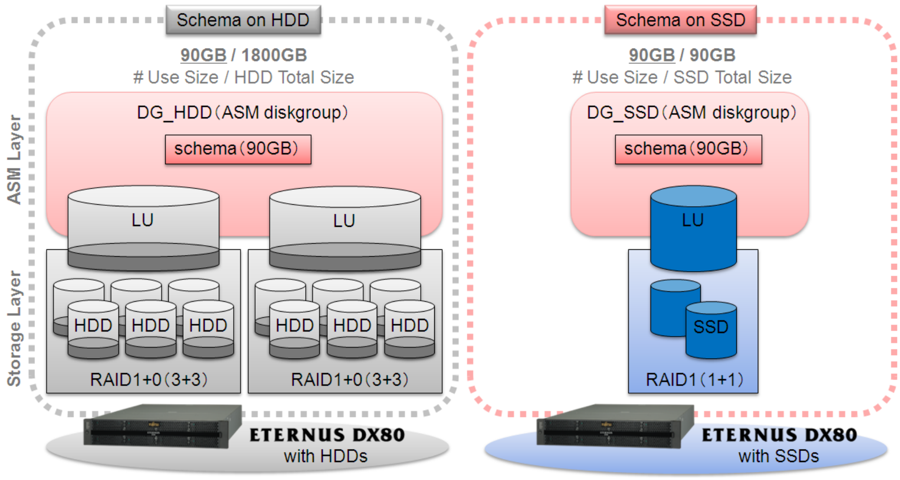



**Fig. 14 Storage configurations with schema on HDDs and on SSDs**

The following graph (Fig. 15) shows the results of the measurement of OLTP processing performance, with the number of threads increased for each configuration.

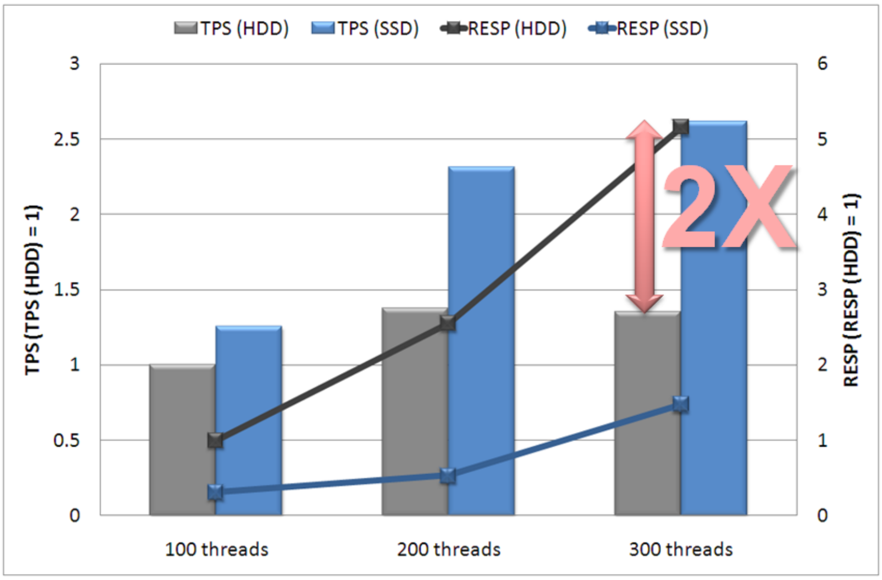



**Fig. 15 OLTP processing performance with schema stored on HDDs and on SSDs.**

Configuration of the schema on SSDs improved the bottleneck in IOPS performance that was generated when the schema was configured on HDDs, and recorded twice the TPS value for HDD at 300 threads. The CPU utilization and the IOPS per drive are shown in Fig. 16.

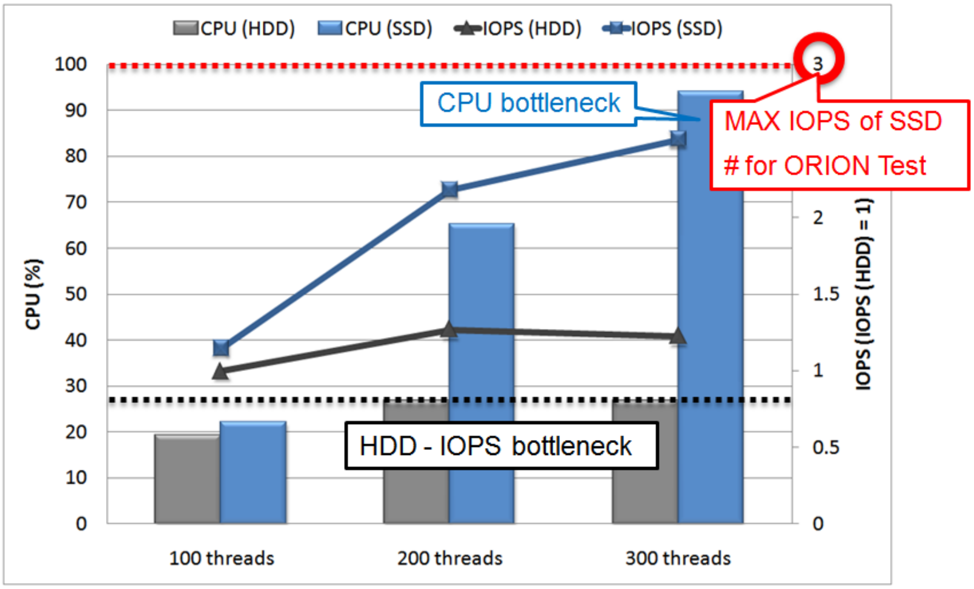



**Fig. 16 OS statistics with schema stored on HDDs and on SSDs**

When the schema was configured on HDDs, the TPS peaked at 200 threads because of the bottlenecked IOPS performance of HDDs. When the schema was configured on SSDs, on the other hand, the IOPS performance of SSDs did not reach its limit even at 300 threads. As a result, the CPU usage reached 100%, causing the TPS to hit the ceiling. In this case, further performance improvement can be achieved by adding nodes in Oracle Real Application Clusters or replacing the database server with a higher-end model.

Based on these results, we confirmed that arranging the schema on SSDs improved the bottleneck in IOPS performance of HDDs and achieved performance improvement in the OLTP system (see Fig. 17).

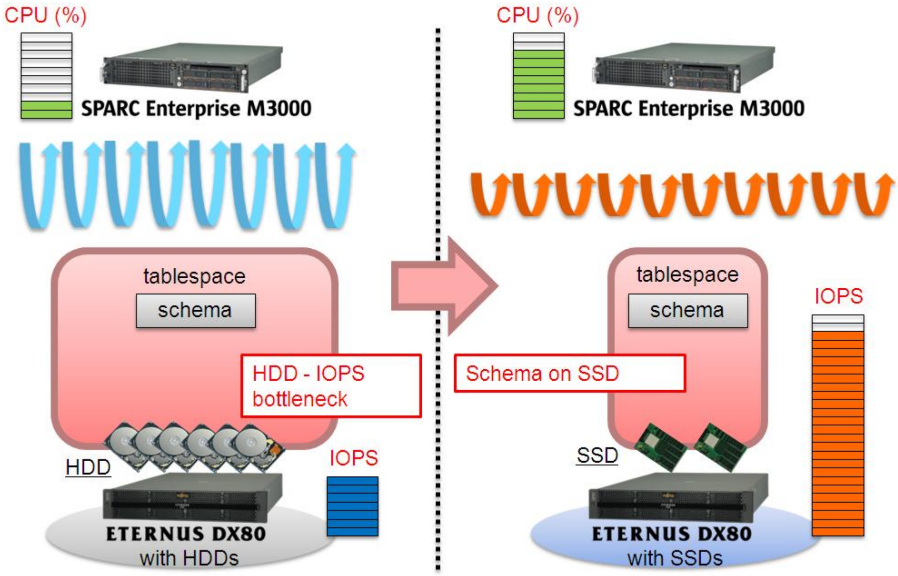



**Fig. 17 Improving OLTP system performance with schema stored on SSDs**

### Selective configuration of schema objects on SSDs

Since the SSD is more expensive than the HDD per unit of capacity, it is impractical to configure an entire schema on SSDs. Even if it is possible to configure the entire schema on SSDs at the time of system construction, it is costly to add SSDs to handle future increases in data volumes. To effectively use the capacity of expensive SSDs, the user can select the specific schema objects to assign to SSDs. Based on the results of measurement using Oracle ORION, we confirmed that an increase in the percentage of writes reduced the maximum value of IOPS. Therefore, in an OLTP system where SELECT SQL is frequently issued, transfer of as many reads as possible from HDDs to SSDs will improve the cost performance.

In the following validation test, schema objects that caused frequent reads (physical reads) from HDDs were selected from the items in “Segments by Physical Reads” in the AWR report and stored on SSDs by priority. Fig. 18 shows the items in “Segments by Physical Reads” in the AWR report, with the schema stored on HDDs.

```
Segments by Physical Reads                     DB/Inst: ORCL/orcl_1   Snaps: 68-69
-> Total Physical Reads:        2,298,069
-> Captured Segments account for   99.8% of Total

           Tablespace                      Subobject  Obj.       Physical
Owner         Name    Object Name            Name     Type          Reads  %Total
---------- ---------- -------------------- ---------- ----- ------------ -------
JPETSTORE  JPETTBS    ITEM                            TABLE      486,565   21.17
JPETSTORE  JPETTBS    PK_ITEM                         INDEX      469,820   20.44
JPETSTORE  JPETTBS    PK_INVENTORY                    INDEX      469,691   20.44
JPETSTORE  JPETTBS    INVENTORY                       TABLE      454,410   19.77
JPETSTORE  JPETTBS    PRODUCT                         TABLE      370,374   16.12
---------- ---------- -------------------- ---------- ----- ------------ -------
```

**Fig. 18 Items in Segments by Physical Reads in AWR report**

Compared to the schema object ITEM, which has the highest physical read frequency, PK_ITEM and PK_INVENTORY were about the same in terms of the number of physical reads, but they differed significantly in object size (see Table 6). To transfer as many physical reads as possible to SSDs, it is necessary to ensure effective use of the SSD capacity by taking the object sizes into consideration.

**Table 6 Sizes of objects with frequent physical reads**

| OBJECT NAME | SIZE (GB) | PHYSICAL READS | PHYSICAL READS/SIZE (GB) |
|---|---|---|---|
| ITEM | 134.27 | 486,565 | 3624 |
| PK_ITEM | 66.35 | 469,820 | 7081 |
| PK_INVENTORY | 66.35 | 469,691 | 7079 |
| INVENTORY | 59.94 | 454,410 | 7581 |
| PRODUCT | 25.24 | 370,374 | 14674 |

The highest effect is likely to be achieved by arranging schema objects with a large number of physical reads per size (“Physical reads/size (GB)” in Table 6) on SSDs. To prevent the degradation of the maximum IOPS value of SSDs, INVENTORY for product inventory management, which is subjected to UPDATE, was excluded. For validation, we arranged three schema objects, PK_ITEM, PK_INVENTORY, and PRODUCT, which accounted for about 57% of all reads on SSDs (see Fig. 19).

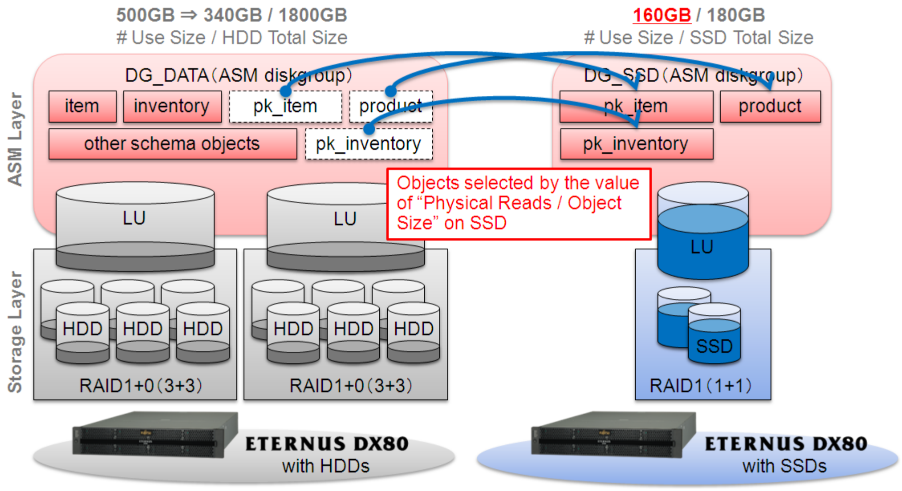



**Fig. 19 Storage configuration with selective configuration of schema objects on SSDs**

The following graph (Fig. 20) shows measurements of OLTP processing performance, with the select range varied by 200 threads for each configuration.

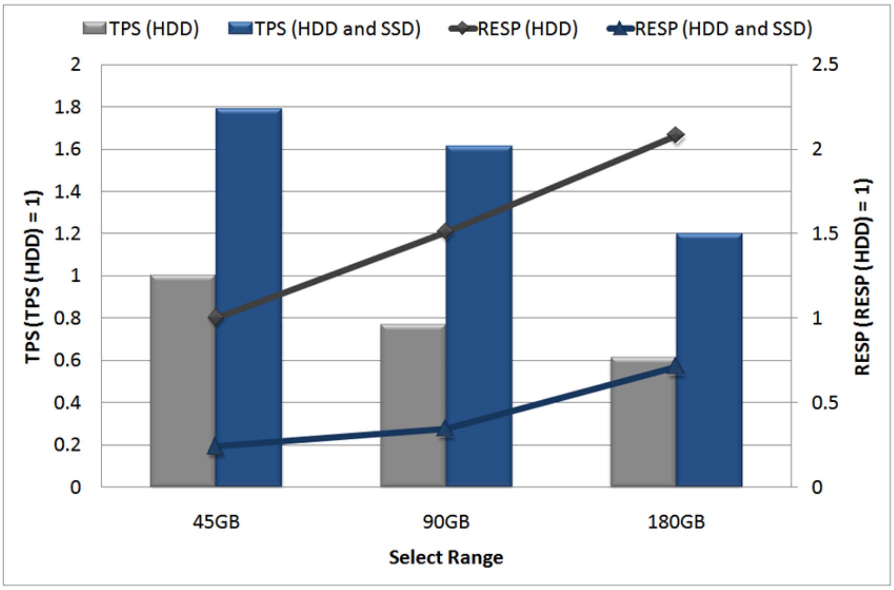



**Fig. 20 OLTP processing performance with selective configuration of schema objects on SSDs**

The results of measurement with selective configuration of schema objects on SSDs indicated that it was possible to improve the bottleneck in HDD IOPS performance and enhance the OLTP system performance by arranging schema objects with a large number of physical reads on SSDs when the entire schema could not be placed on SSDs (see Fig. 21).

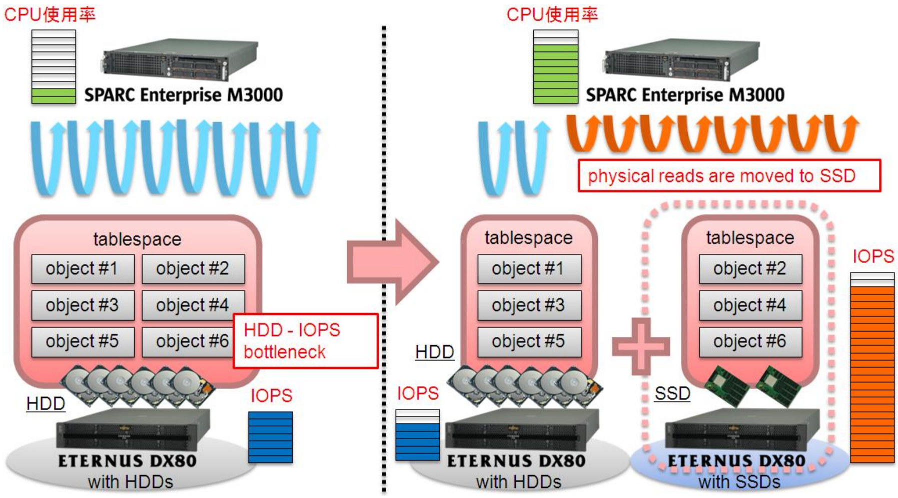



**Fig. 21 Improving OLTP system performance with selected configuration of schema objects on SSDs**

The following graph (Fig. 22) shows the CPU usage and IOPS per drive in each measurement.

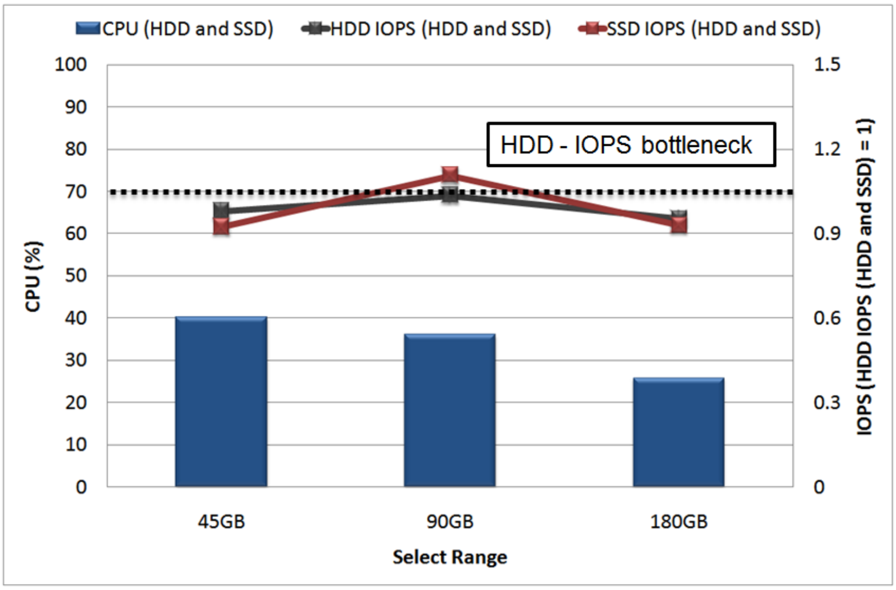



**Fig. 22 OS statistics with selective configuration of schema objects on SSDs**

Even though approximately 57% of all physical reads were transferred to SSDs, the bottleneck in HDD IOPS performance was not solved, causing the OLTP processing performance to peak out. Generally, since the frequency of access to data in schema objects is not evenly distributed within the schema objects, the schema objects stored on SSDs contain data blocks with low access frequency (see Fig 23). For maximum use of SSD capacity, it is necessary to configure as many data blocks accessed at high frequency as possible on SSDs.

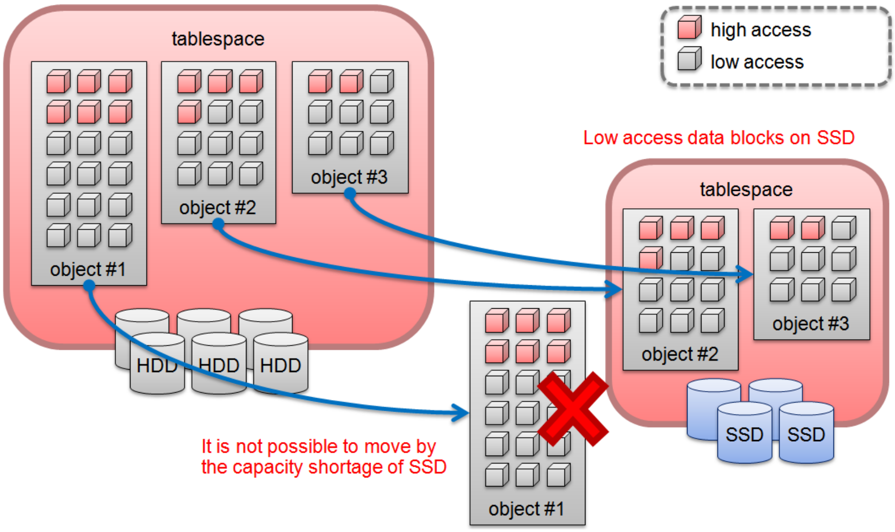



**Fig. 23 Mixture of data blocks accessed at high frequency and data blocks with low access frequency**

## Approach for more effective use of SSD capacity

The following two approaches may be used to configure as many data blocks accessed at high frequency as possible on SSDs for more effective use of SSD capacity, although these techniques were not tested in our validation experiments.

(1) Information Lifecycle Management (ILM) using Oracle Partitioning

By partitioning objects with a large number of physical reads and arranging only the partitions with high access frequency on SSDs, the SSD capacity can be utilized more effectively. This transfers more physical reads from HDDs to SSDs, thus improving the OLTP system performance.

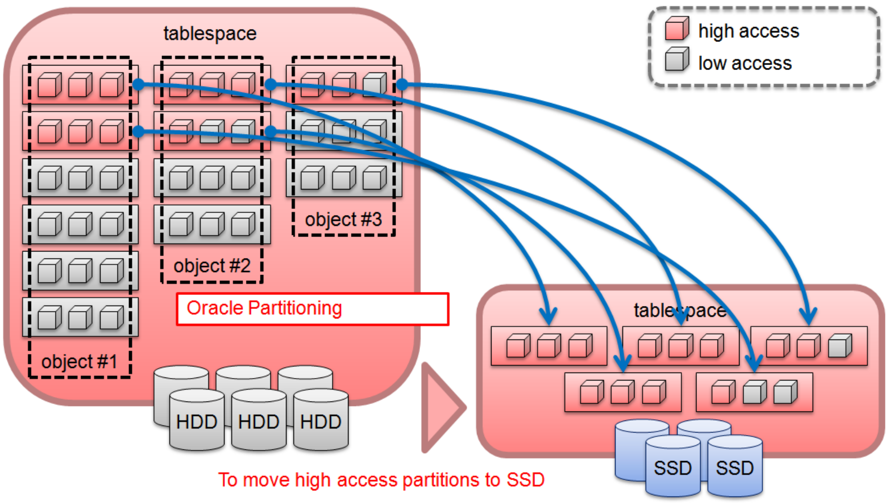



**Fig. 24 ILM using Oracle Partitioning**

(2) Advanced compression — Employment of OLTP table compression

By compressing table objects to smaller sizes, the number of objects that can be placed on SSDs increases, thus enabling the transfer of more physical reads from HDDs to SSDs. Additionally, it also reduces the number of data blocks that must be accessed during query due to an increased number of records in each data block, thereby improving OLTP performance.

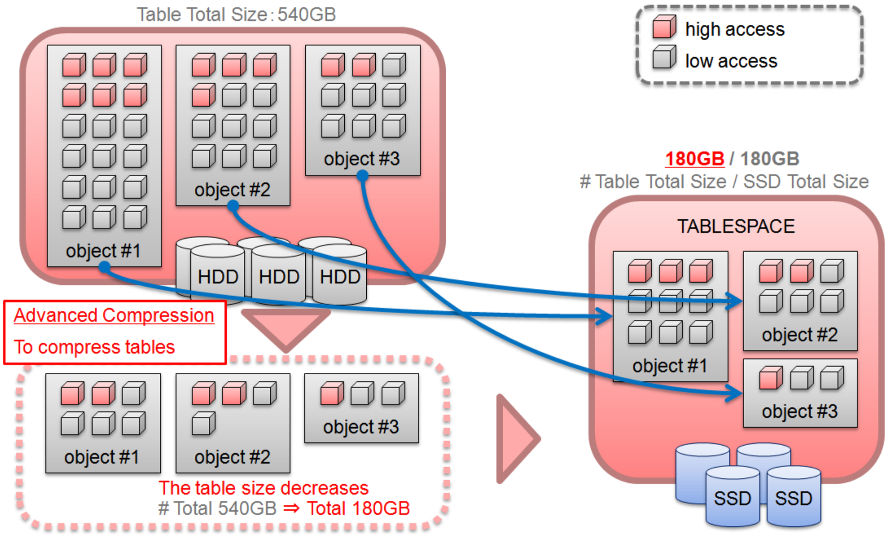



**Fig. 25 Advanced compression -- Employment of OLTP table compression**

Regarding the effectiveness of advanced compression, refer to the following reference:

Verification of Oracle Database 11g for Data Warehousing Using Fujitsu SPARC Enterprise

(http://www.fujitsu.com/global/services/computing/server/sparcenterprise/key-reports/featurestory/sparce-feature081216.html)

## Summary on the use of SSDs in place of HDDs

The OLTP system performance can be improved by arranging the schema on SSDs. However, since it is generally expensive to place the entire database schema on SSDs, it is necessary to select schema objects to be stored on SSDs. In the validation tests described in this paper, schema objects with a large number of reads from HDDs were stored on SSDs. The results of the tests confirmed the improvement of the OLTP system performance, but they also indicated that the SSDs failed to provide the highest cost performance since data blocks with low access frequency were also placed on the SSDs.

### Performance achieved with Database Smart Flash Cache

What are described below are the performance improvements and advantages achieved by using Database Smart Flash Cache in an OLTP system.

#### Improving OLTP system performance achieved by using Database Smart Flash Cache

As shown in Fig. 26, SSDs mounted in the ETERNUS DX80 were added as a Database Smart Flash Cache area for the database system configured with 12 HDDs.

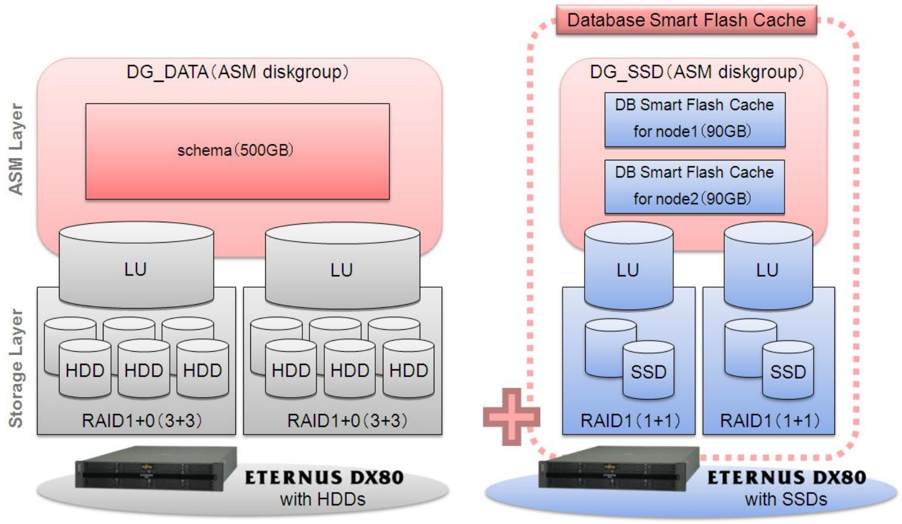



**Fig. 26 Storage configuration using Database Smart Flash Cache**

A Database Smart Flash Cache area is required for each node in a RAC environment. Table 7 shows the SSD configuration used in the validation test and the size of the Database Smart Flash Cache area for each node.

**Table 7 Number of LUs in Database Smart Flash Cache area and size per node**

| RAID CONFIGURATION | NO. OF LUS | TOTAL CAPACITY (GB) | SIZE (GB) OF DATABASE SMART FLASH CACHE AREA PER NODE |
|---|---|---|---|
| RAID1 (1+1) | 2LU | 180 | 90 |

To confirm the change in the OLTP processing performance trend in relation to data volume and Database Smart Flash Cache area size, the search range (Select Range) was varied in the validation test using an application that controlled the range of data to be accessed. The following graph (Fig. 27) shows the results of measurement.

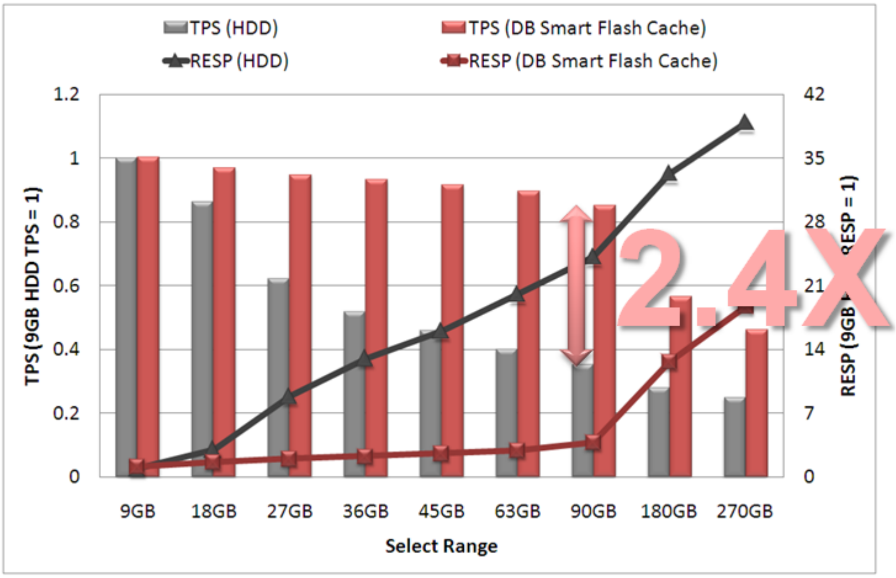



**Fig. 27 Improving OLTP processing performance achieved by using Database Smart Flash Cache**

Based on these results, we confirmed that the use of Database Smart Flash Cache could resolve the bottleneck in HDD IOPS performance and increase the OLTP system performance. When the search range was 9 GB, the buffer cache hit rate was nearly 100%; thus, there was no difference in the OLTP processing performance between the tested systems. When the system was configured only with HDDs, a wider search range increased the rate of accessing data blocks on HDDs, thus significantly degrading the OLTP processing performance. On the other hand, when Database Smart Flash Cache was used, the degradation of the OLTP processing performance resulting from a larger search range was very gradual.

The following graph (Fig. 28) shows the HDD and SSD I/O tendency in each configuration.

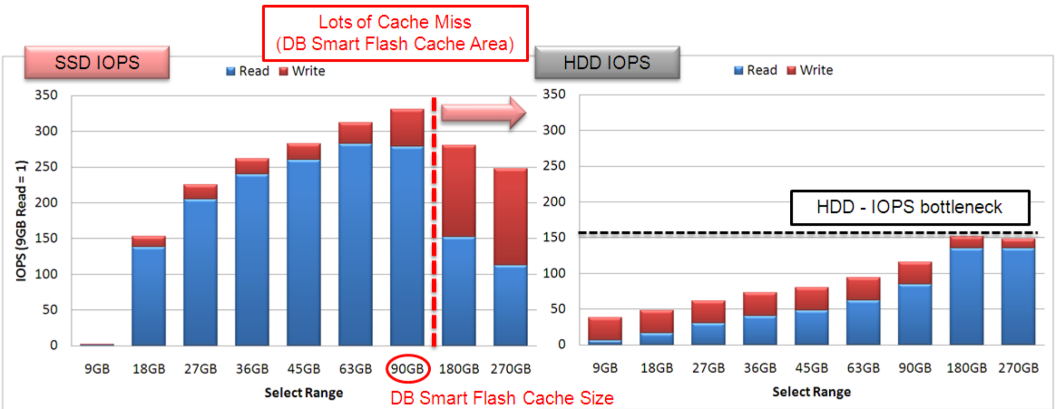



**Fig. 28 HDD and SSD I/O trends**

If a cache miss occurs in the Database Smart Flash Cache area, HDD I/O takes place. Since the retrieved data blocks are placed in the buffer cache, data blocks with lower access frequency are written to the Database Smart Flash Cache area. When the search range expands beyond the size of the Database Smart Flash Cache area, the HDD I/O frequency and the ratio of writes to SSDs increase. Therefore, the effectiveness (difference in performance from the HDD-only configuration) of Database Smart Flash Cache is expected to become the greatest when the search range approaches the size of the Database Smart Flash Cache area.

The following graph (Fig. 29) shows the average CPU usage (%usr + %sys) in each configuration.

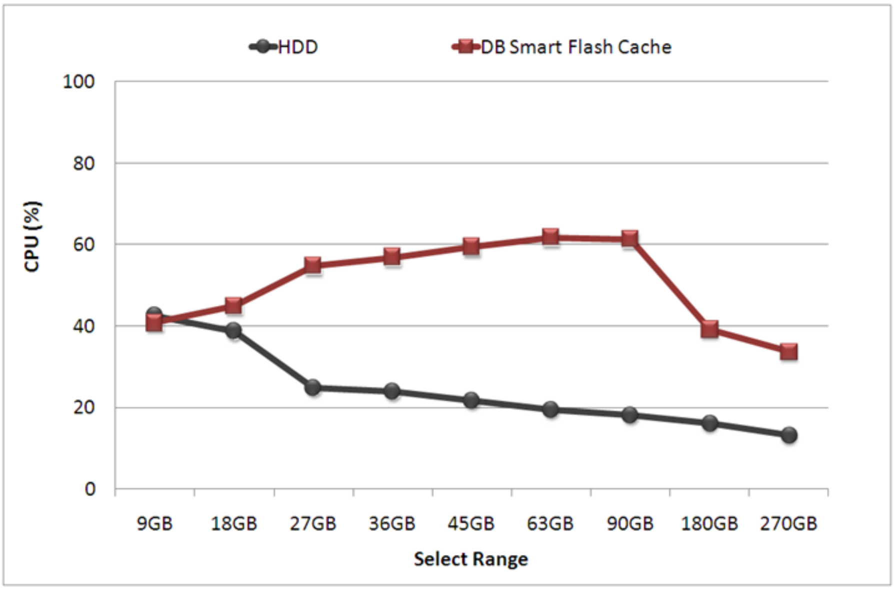
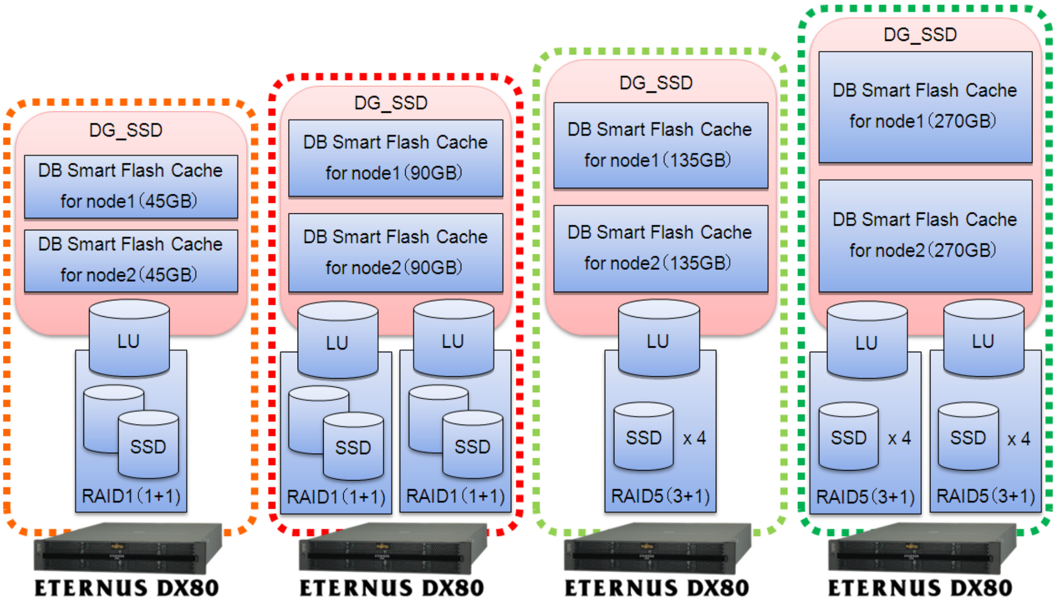



**Fig. 29 CPU usage when Database Smart Flash Cache is used**

In the configuration with only HDDs, an increase in the search range resulted in a bottleneck in HDD IOPS performance, reducing the TPS. This caused the CPU usage to decrease. On the other hand, when Database Smart Flash Cache was used, the TPS degraded slightly when the search range increased, but the CPU usage increased. Even if the expanded search range was within the size of the Database Smart Flash Cache area, this increase reduced the buffer cache hit rate, increasing the frequency of I/O to and from SSDs (see Fig. 28). It is suspected that the increase in the CPU usage was caused by the increased use of CPU resources by I/O system calls from OS kernels and by the use of Database Smart Flash Cache.

### Performance achieved by the increased size of Database Smart Flash Cache area

In the validation test described above, an inadequate size of the Database Smart Flash Cache area resulted in a tendency for OLTP processing performance degradation. If this problem occurs, the OLTP processing performance can be improved by increasing the size of the Database Smart Flash Cache area. In the following validation test, the RAID level and the number of LUs used were varied to increase the size of the Database Smart Flash Cache area (see Fig. 30).




| RAID CONFIGURATION | NO. OF LUS | TOTAL NO. OF SSDS | TOTAL CAPACITY (GB) | SIZE (GB) OF DATABASE SMART FLASH CACHE AREA PER NODE | CONFIGURATION NAME |
|---|---|---|---|---|---|
| RAID1 (1+1) | 1LU | 2 | 90 | 45 | RAID1 (1+1) x 1 |
| RAID1 (1+1) | 2LU | 4 | 180 | 90 | RAID1 (1+1) x 2 |
| RAID5 (3+1) | 1LU | 4 | 270 | 135 | RAID5 (3+1) x 1 |
| RAID5 (3+1) | 2LU | 8 | 540 | 270 | RIAD5 (3+1) x 2 |

**Fig. 30 SSD configurations of Database Smart Flash Cache areas and size of each node**

Fig. 31 shows the results of measurement, with the search range varied for each configuration.

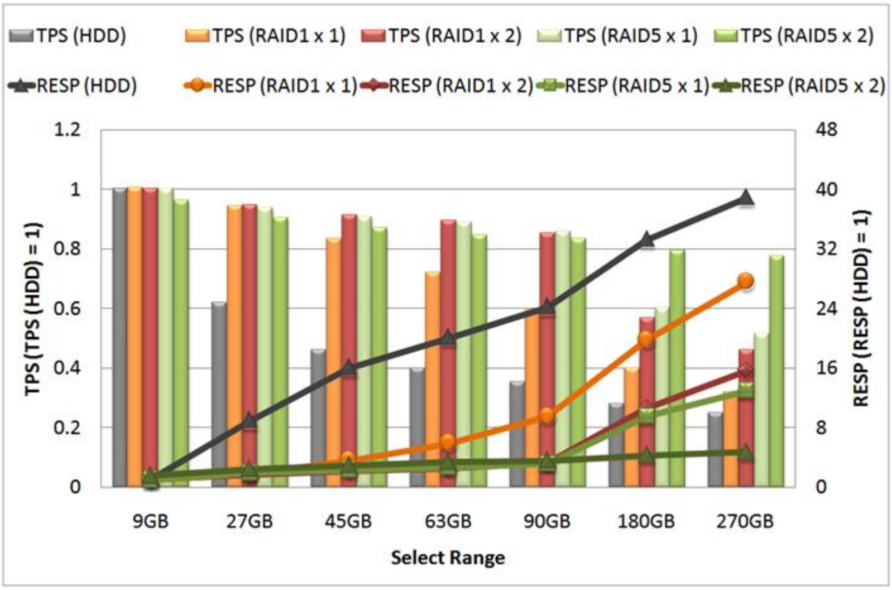



**Fig. 31 Improving OLTP processing performance achieved by the increase in Database Smart Flash Cache area size**

We confirmed that increasing the size of the Database Smart Flash Cache area prevented the degradation of OLTP processing performance caused by the expanded search range. Therefore, it is possible to cope with an increase in data volume by adding SSDs to expand the Database Smart Flash Cache area according to the search range.

When Database Smart Flash Cache is used, an administrative area is secured in the buffer cache. The size of the administrative area is about 2 to 3% of the Database Smart Flash Cache area. In the validation test, a slight effect was observed in the “RAID5 x 2” configuration when the search range was within the size of the Database Smart Flash Cache area (see Fig. 31). The buffer cache in the validation environment was 16 GB in size. When the size of the Database Smart Flash Cache area was 270 GB, an administrative area of 5 to 8 GB, which is 2 to 3% of 270 GB, was allocated in the 16-GB buffer cache. As a result, the remaining buffer cache area could not cache the data in the search range, thus causing a slight decrease in the TPS.

### Comparison of performance of different SSD usage methods

In this section, the results of measurement described in the sections, “Selective configuration of schema objects on SSDs” and “Improving OLTP system performance achieved by using Database Smart Flash Cache,” are compared. To match the SSD configuration described in the former section, SSDs were used as 2 LUs in RAID1 (1+1) to configure a Database Smart Flash Cache area and the size of the Database Smart Flash Cache area for each node was set to 90 GB. The following graph (Fig. 32) shows the results of measurement.

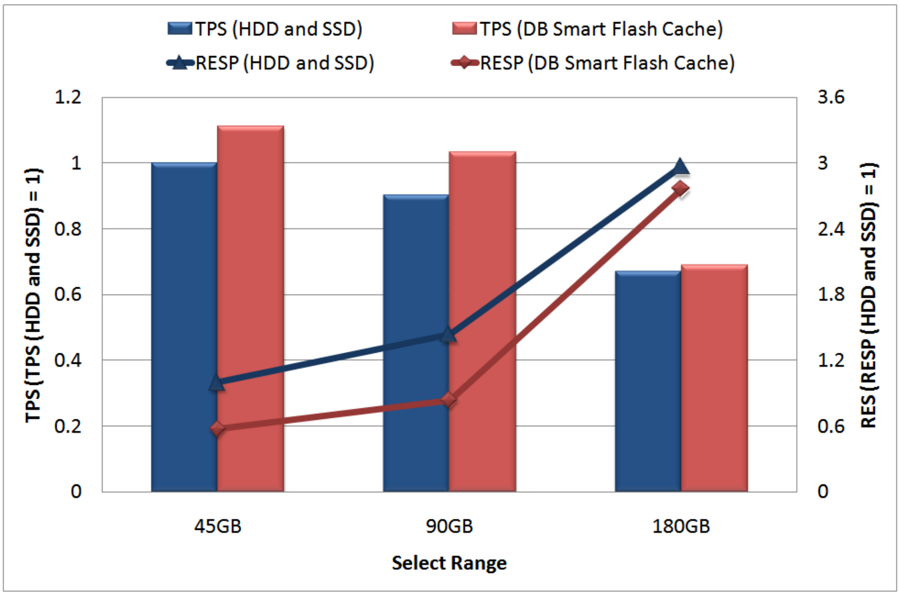



**Fig. 32 Improving OLTP processing performance achieved by different SSD usage methods**

In the validation test, the use of Database Smart Flash Cache achieved higher OLTP processing performance. When objects were selectively stored on SSDs, the bottleneck in HDD IOPS performance remained, causing the OLTP processing performance to peak out. By comparison, the use of Database Smart Flash Cache improved the bottleneck in HDD IOPS performance, allowing effective use of CPU resources (see Fig. 33).

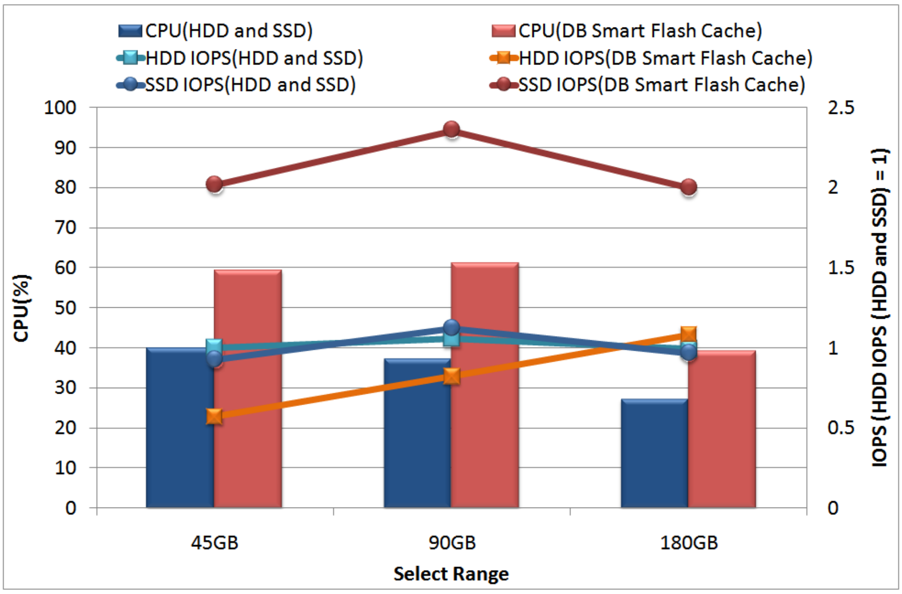



**Fig. 33 OS statistics of different SSD usage methods**

### Summary on the use of Database Smart Flash Cache

Database Smart Flash Cache uses SSDs as a database cache. Data blocks that are not retained in the buffer cache even though they are frequently accessed are automatically placed in the Database Smart Flash Cache area. This enabled us to confirm the improvement of OLTP system performance achieved by maximum use of SSDs, without performing analysis or operation similar to those we performed when SSDs were used in place of HDDs.

However, the following points require your attention.

- Administrative area for Database Smart Flash Cache area is allocated in the buffer cache.
- CPU overhead can result from the use of Database Smart Flash Cache
- Database Smart Flash Cache area is necessary for each node in an RAC environment.

## Summary

While many OLTP systems currently suffer from inadequate I/O performance of storage devices, SSD's offering high-speed I/O capability can provide a solution to those problems and improve the performance of the OLTP systems. The validation tests described in this paper confirm the advantages of using SSDs for the purpose of improving system performance. Thus, we were able to propose an effective solution to problems in OLTP systems.

When SSDs are used in place of HDDs, it is impractical to configure all data on SSDs since reduction of IT cost is a major issue for most companies. Therefore, it is necessary to analyze the databases, and then design the storage configuration and arrange the data according to the access frequency and service level. However, as seen in the results of the validation tests described herein, even selective configuration of data on SSDs does not result in maximum use of the SSD performance and may prevent expected performance improvements from being attained.

When Database Smart Flash Cache is used, OLTP processing performance can be improved by using SSDs as an extended area of the buffer cache. Since data will be automatically stored on SSDs, it eliminates the need for the analysis of databases and reconfiguration of data. This is an excellent solution because it provides high OLTP processing performance while minimizing the installation and operating costs.

Fujitsu ETERNUS storage systems, including the DX80 used in the validation tests as well as entry-level models, can mount SSDs, thus allowing the use of SSDs at a relatively low cost. When combined with SPARC Enterprise that supports Database Smart Flash Cache, the ETERNUS system enables the configuration of a high-speed database system with a minimum installation cost.

# Appendix A

## Tuning Database Smart Flash Cache

Although data blocks accessed at high frequency are automatically arranged in the Database Smart Flash Cache area, it is also possible for the user to select what data to cache in preference to other data and what data not to cache. Described below are the corresponding procedures and consequences.

The following is an example of the procedure. Using the STORAGE clause, specify KEEP, NONE, or DEFAULT for each schema object.

- Example of setting for caching schema objects by priority

```
ALTER TABLE <OBJECT_NAME> STORAGE (FLASH_CACHE KEEP);
```

- Example of setting for not caching schema objects

```
ALTER TABLE <OBJECT_NAME> STORAGE (FLASH_CACHE NONE);
```

- Example of default setting

```
ALTER TABLE <OBJECT_NAME> STORAGE (FLASH_CACHE DEFAULT);
```

At the next step, Database Smart Flash Cache is tuned to resolve bottlenecks in SSD IOPS performance.

The section, "Improving OLTP system performance achieved by using Database Smart Flash Cache," described a RAID1 configuration in which SSDs were used as 1 LU for the Database Smart Flash Cache area. In this configuration, the highest SSD IOPS was recorded when the search range was 45 GB (see Fig. 28). When the number of application threads increased from 200 to 300 in this configuration, the SSD IOPS performance became a bottleneck.

In this validation experiment, in the above-mentioned environment, the setting was made to disable caching of INVENTORY for tables subject to UPDATE. When UPDATE is issued, the corresponding data block in the Database Smart Flash Cache area becomes invalid. Therefore, the setting that prevents that schema object from being cached can improve the efficiency of the use of the Database Smart Flash Cache area.

The TPS and response time of each configuration and the average response times[7] of HDDs and SSDs are shown below (Figs. 34 and 35).

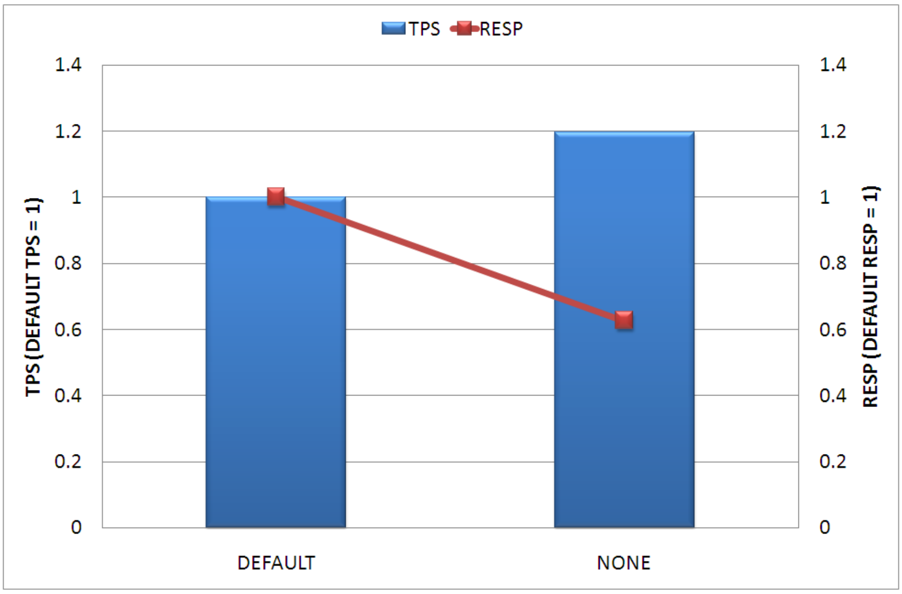



**Fig. 34 Improving OLTP processing performance achieved by tuning**

[7] The average wait time of one execution of database wait events, "db file sequential read" and "db flash cache single block physical read," was used for each. Since "db file sequential read" expresses the wait time for the acquisition of a data block from HDD, it was regarded as an HDD response time from the perspective of the database. Since "db flash cache single block physical read" expresses the wait time for the acquisition of a data block from the Database Smart Flash Cache area, it was considered as the SSD response time from the perspective of the database.

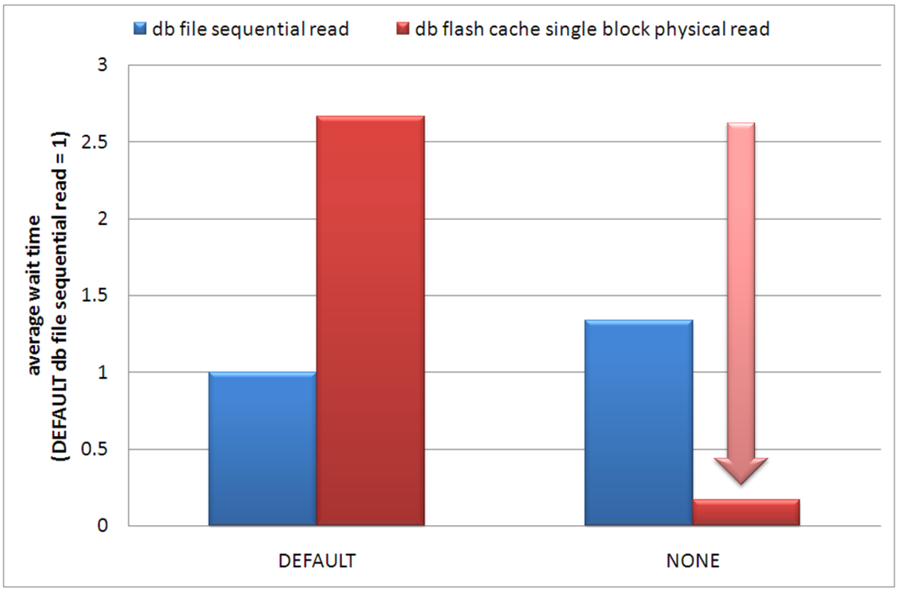



**Fig. 35 Improving SSD response times**

We confirmed that tuning Database Smart Flash Cache resolved the bottleneck in SSD IOPS performance and improved OLTP processing performance. With the default setting, SSD response times were significantly slower than HDD response times. However, disabling INVENTORY caching increased HDD loads slightly but improved SSD response times dramatically.

As described above, Database Smart Flash Cache is equipped with a function that enables the control of cache operations for each schema object. This function allows the user to perform tuning based on I/O trends to improve the efficiency of the SSDs and achieve higher OLTP system performance.

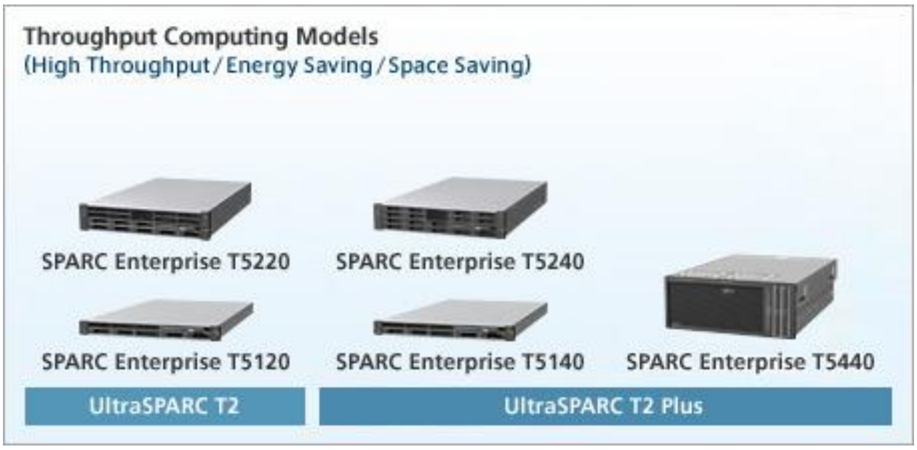
## Appendix B

### Overview of SPARC Enterprise

Featuring a SPARC/Solaris architecture, the SPARC Enterprise UNIX servers were developed to offer enhanced functions and high performance. The SPARC Enterprise series offers the following models to match diverse business characteristics:

- High performance, high reliability, and high extendibility server for use across a wide range of business applications, including databases and batch processing


- Server suitable for Web front ends, on-line transaction processing, and other applications




For details, go to the following website:

www.fujitsu.com/sparcenterprise

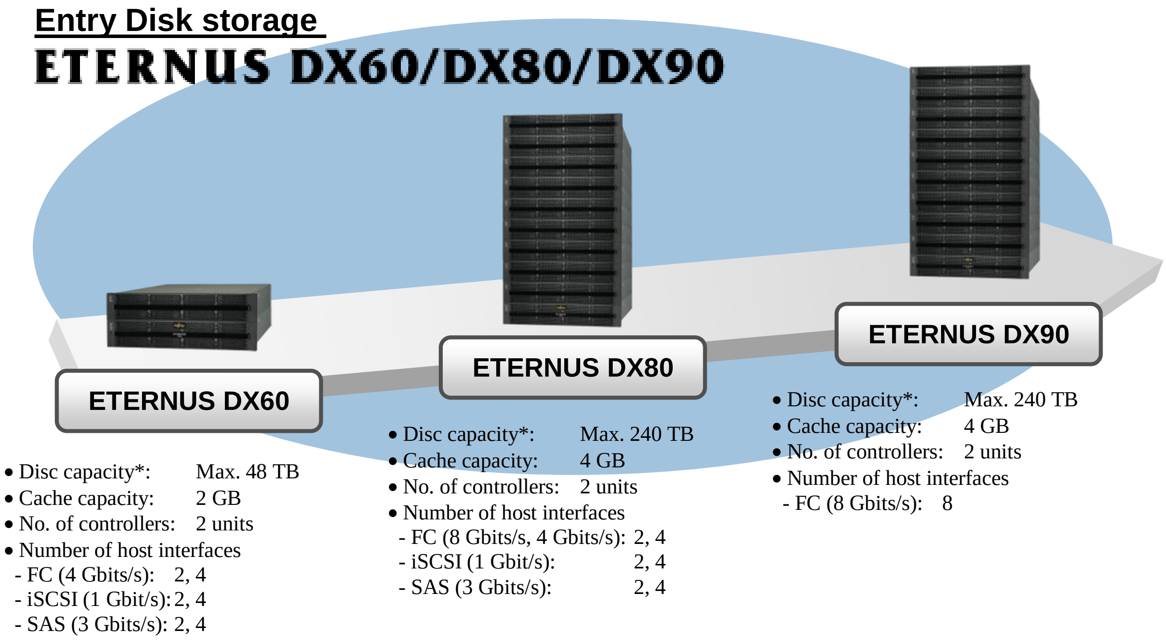
## ETERNUS DX series

The ETERNUS DX series products are high-reliability, high performance disk arrays incorporating RAID technology. The lineup includes the following products:

- The ETERNUS DX60/DX80/DX90 are compact, cost-effective entry-level disk arrays suitable for small- and mid-sized companies and distributed system environments.
- The ETERNUS DX400 series mid-range disk arrays are excellent solutions for responding to various challenges, including meeting compliance demands, and offer high cost performance.
- The ETERNUS DX8000 series enterprise disk arrays are excellent solutions for compliance, system integration, and various other issues.

### ETERNUS DX60/DX80/DX90

These entry-level disk arrays can be used as backbone IA/UNIX/PC servers. The compact units feature high extendibility and reliably and provide optimal solutions for small- and mid-sized systems. The ETERNUS DX90 even supports the remote advanced copy function offered by higher-end models.




* The above capacities indicate figures for physical capacity (including system disk and hot spare disk) calculated based on the assumption that 1 kByte = 1,000 Bytes.
When mounting 2 TB nearline SAS disk drives

### ETERNUS DX400 series

The ETERNUS DX400 series mid-range disk arrays are suitable for use as backbone IA/UNIX/PC servers. This series offers a wide range of products with excellent extendibility, solutions geared to meeting wide-ranging information system requirements, including data integrity and compliance.


**ETERNUS DX410**

- Disc capacity*: Max. 414 TB
- Cache capacity: 8 GB
- No. of controllers: 2 units
- Number of host interfaces
  - FC (8 Gbits/s, 4 Gbits/s): 2 to 8
  - iSCSI (1 Gbit/s): 2, 4

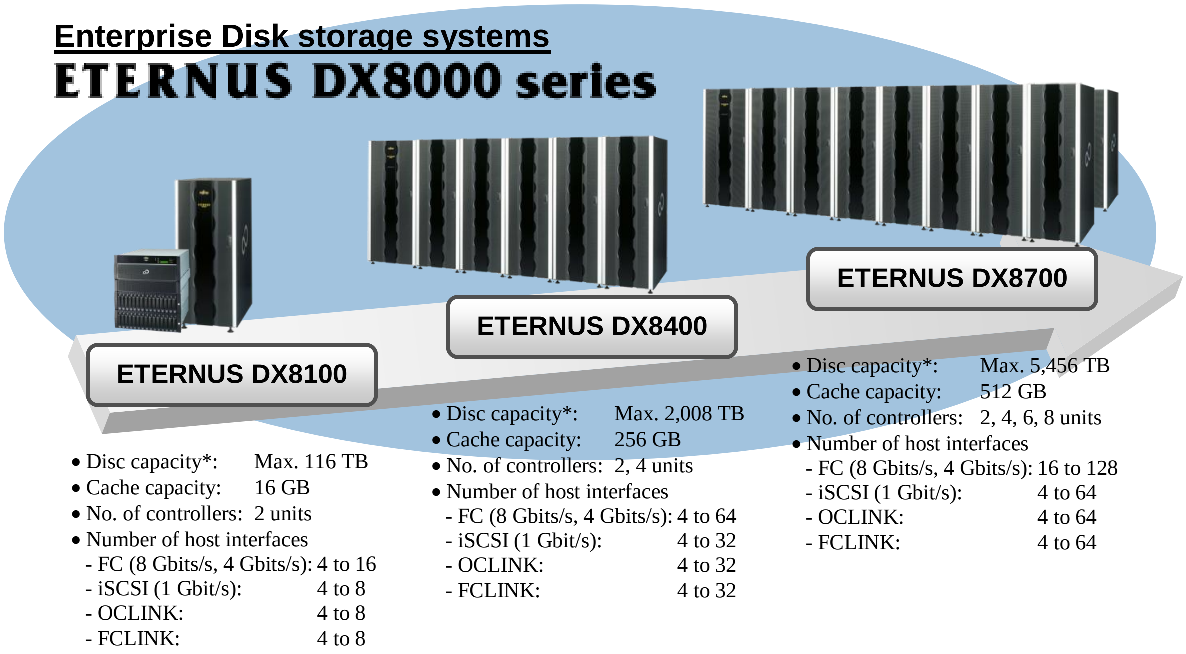
**ETERNUS DX440**

- Disc capacity*: Max. 834 TB
- Cache capacity: 32 GB
- No. of controllers: 2 units
- Number of host interfaces
  - FC (8 Gbits/s, 4 Gbits/s): 4 to 16
  - iSCSI (1 Gbit/s): 4, 8

* The above capacities indicate figures for physical capacity (including system disk and hot spare disk) calculated based on the assumption that 1 kByte = 1,000 Bytes.
When mounting 2 TB nearline SATA disk drives

### ETERNUS DX8000 series

The ETERNUS DX8000 series enterprise disk arrays are suitable for multi-platform applications ranging from global servers to backbone IA/UNIX/PC servers. They support various system environments and realize large-scale system integration at the enterprise level.




* The above capacities indicate figures for physical capacity (including hot spare disk) calculated on the assumption that 1 kByte = 1,000 Bytes.
When mounting 2 TB nearline SATA disk drives

Oracle Database 11g Release 2
Improving OLTP System Performance by Using Database Smart Flash Cache with Fujitsu SPARC Enterprise
April 2010



Fujitsu Limited
Shiodome City Center
1-5-2, Higashi-Shimbashi, Minato-ku, Tokyo

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



Author: Tomohiro Iwamoto

Oracle Corporation Japan
Oracle Aoyama Center
2-5-8, Kita-Aoyama, Minato-ku, Tokyo