White Paper

High-Speed Flash Storage Provides Accurate Estimations of Processing Times

The flash storage system offers a high level of responsiveness until it reaches its saturation point.
By knowing this saturation point, we can estimate the maximum performance of the SQL database.
This document describes the characteristics of flash storage and how to estimate the SQL processing time of a database.

Table of Contents

List of Figures

List of Tables

Preface

Recent storage systems employ solid state drives (SSD) that use flash memory in addition to hard disk drives (HDD). Because the price difference between SSDs and HDDs is decreasing, SSDs are gradually being used for data warehouses (DWH), which handle large amounts of data.
This document describes the characteristics of SSDs and how to estimate the SQL processing times of databases.

The product lineup and product information stated in this document are current as of November 2019.

■ Target Readers

This document targets the following audience:

- Those who are considering the installation or replacement of a storage system for a database (especially for a DWH)

■ Required Knowledge

This document assumes that the audience has the following knowledge:

- General knowledge of file systems and storage systems
- General knowledge of databases

■ Applicable Models

The descriptions in this document are based on the following ETERNUS AF series All-Flash Arrays:

- FUJITSU Storage ETERNUS AF150 S3, AF250 S3/S2, and AF650 S3/S2 All-Flash Arrays

■ Naming Conventions

The following abbreviation is used in this document:

- FUJITSU Storage ETERNUS AF150 S3, AF250 S3/S2, and AF650 S3/S2 All-Flash Arrays .... ETERNUS AF series

■ Terminology

The following terms are used in this document:

- Storage device ....Device such as an HDD, SSD, magnetic tape, or an optical disk that can store data
- Storage system ....Storage system consisting of HDDs and SSDs
- Flash storage ....Storage system, such as the ETERNUS All-Flash Array that consists of SSDs
- Database buffer cache ....Memory areas in the server that are separately allocated to each database
- Cache hit ....Required data for database processing is stored in the database buffer cache and can be immediately read from the database buffer cache
- Cache miss ....Required data for database processing is not stored in the database buffer cache so the storage device, such as HDDs and SSDs, is accessed

■ Units

The following units are used in this document:

- IPOs ....I/O access count to a storage device per second
- MB/s ....The amount of data transfer per second (1MB/s = 1,000,000 byte/s)

1 How to Increase the Speed of Computer Systems

Computer systems have always been required to process faster and process more data. This requirement is fulfilled mainly in two ways: by "increasing the number of operations" and by "achieving higher parallelization".

"Increasing the number of operations" means the increase of operations per unit time to execute more operations, although different units are used depending on the target devices. Common examples include the use of a faster clock speed CPU and the use of a faster RPM HDD.

"Achieving higher parallelization" increases the amount of data handled by one operation. Common examples include adding CPU cores, increasing memory capacity, and adding HDDs. Increasing the number of cores from one to ten logically increases performance ten times more than the original level.

To speed up computer systems, in addition to CPUs and memory, data storage devices (SSDs and HDDs) are just as important. When the data unit or amount is small, CPUs and memory can achieve the required performance. However, because a DWH handles large amounts of data, the performance of the storage device must be considered.

The following sections explain how to compare the numbers of operations per second of the CPU, memory, SSD, and HDD – all of which constitute a computer system – to find out the difference in performance. Furthermore, the data transfer amount per second is calculated from the number of operations per second and then the results are compared.

1.1 Comparison of the Number of Operations

Hz (hertz), which is used for CPUs and memory, describes the operation frequency (cycle). Performing one operation per second is expressed as 1Hz. With contemporary computer systems, CPUs operate at 2GHz or higher and memory at about 300MHz. HDDs, on the other hand, use ms (millisecond) as the unit to describe one access time. HDDs operate at about 20ms.

This section shows how to convert these numbers to the numbers of operations per second for comparison:

Because Hz is used for CPUs and memory, the numbers can be directly converted to the number of operations. For a CPU, 2GHz means 2,000,000,000. For memory, 300MHz means 300,000,000.

For an SSD, the number of operations per second is about 50,000. For an HDD, the number of operations per second is 50 because "it takes 20ms for each access attempt".

As shown in the following figure, there is a 6.7-fold difference between the CPU and memory, a 6,000-fold difference between the memory and SSD, and a 1,000-fold difference between the SSD and HDD. The difference between the memory and SSD is especially large.

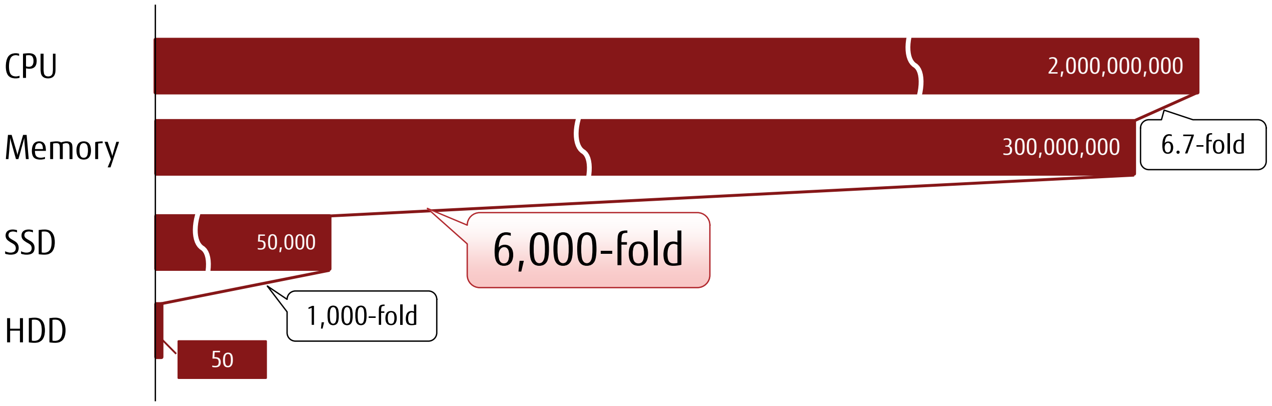


Figure 1-1 Comparison of the Numbers of Operations per Second

1.2 Comparison of the Amount of Data Transfer

This section shows how to compare the amount of data transfer per second. The amount of data transfer per second is calculated by multiplying the data transfer amount of a single operation by the number of operations per second. The data transfer amount of a single operation is normally 64 bits (8 bytes) for the memory and 4KB for the SSD and HDD. The numbers in "1.1 Comparison of the Number of Operations" are used as the number of operations per second.

Amount of data transfer of a single operation × number of operations per second = amount of data transfer per second

Memory:	8 bytes (64 bits)	×	300,000,000 =	2,400,000,000 bytes (2.2GB)
SSD:	4 kilobytes	×	50,000 =	204,800,000 bytes (195MB)
HDD:	4 kilobytes	×	50 =	204,800 bytes (200KB)

The following figure shows a comparison of the memory, SSD, and HDD. The difference of the number of operations per second between the memory and the SSD is 6,000-fold, but the difference of data transfer amount is 12-fold, which is significantly smaller.

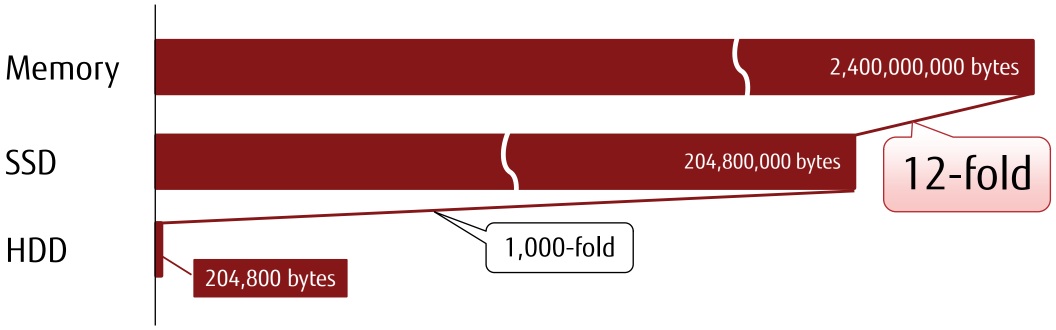


Figure 1-2 Comparison of Amount of Data Transfer per Second

1.3 SSD Is Effective for Accelerating Computer Systems

For accelerating a computer system, it may be effective to use CPUs with faster clock speeds, more CPU cores, or more memory. But these options are expensive and there are limitations on the number of installed devices. So it is difficult to increase these resources. The SSD can deliver performance similar to the memory in systems where large amounts of data are transferred per second. The SSD also has better expandability compared to the CPU and memory.

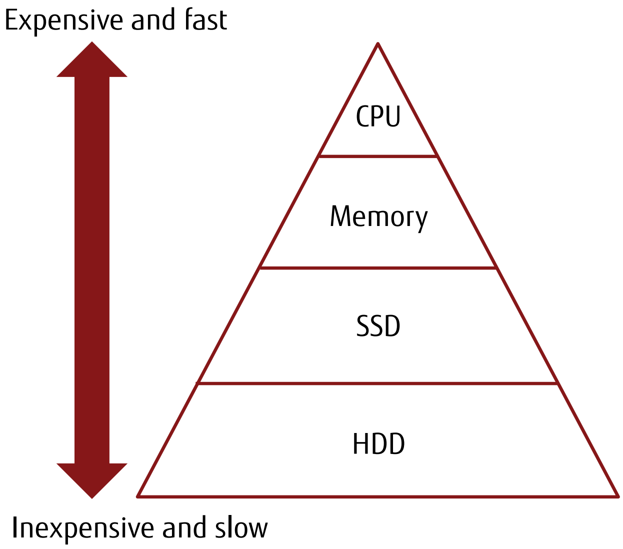


Figure 1-3 Device Layer

When the price of a 2TB memory is compared with the price of a 2TB SSD, the SSD price is 1/48 the memory price, which is very inexpensive. The SSD has a capacity 62 times greater than the capacity of the memory for the same price.

SSDs have higher capacities and lower prices compared to memory and achieve a high-speed, highly-cost-effective computer system.

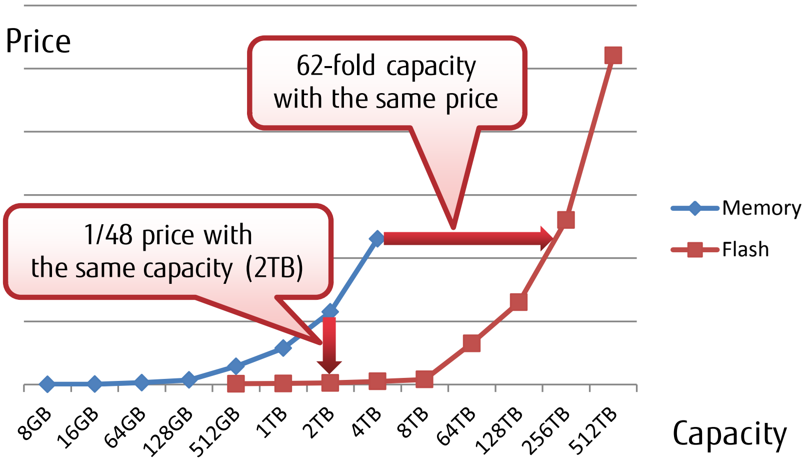


Figure 1-4 Comparison of Price and Capacity between Memory and SSD

2 Performance of HDD Storage Systems

Before the advent of SSDs, main storage devices were HDDs and tapes, which work based on the movement of magnetic media. The performance of such storage systems is difficult to determine because different performance indicators are used depending on the access pattern and the block size. This section explains how performance values are indicated and describes general characteristics of HDD performance.

2.1 Access Patterns and Block Sizes

The access pattern and the block size are important elements for indicating performance of storage systems. This section explains access patterns and block sizes.

2.1.1 Access Patterns

For access patterns, sequential access and random access are used.

- Sequential access

Sequential access involves almost no seek time for magnetic media and almost no rotation wait time, so the time to access the magnetic media accounts for most of the total time.

Sequential access performance is indicated by how much data can be transferred.

The unit of performance is throughput (MB/s), which is the amount of data transfer (MB) per second.

An HDD with a high throughput is often said to "have good sequential access performance".

- Random access

Random access involves frequent movement of the magnetic media to access data that is stored in non-contiguous areas as well as respond to I/O access requests issued for the random data. Random access results in frequent seek times and rotation wait times.

Random access performance is indicated by how many I/O access requests can be processed.

The unit of performance is IOPs (I/O access count per second).

An HDD with a high IOPs value is often said to "have good random access performance".

2.1.2 Block Sizes

The block size is the size of data being transferred at a time when a storage device is accessed. A storage device is always accessed in units of blocks. Data transfer in units of blocks is a method of data transfer that is common in computing and not limited to storage devices. In this document, however, a block size means the data length or the average data length that user applications request at a single I/O access attempt.

Block size is determined by the operating system or the user application, and is basically invisible to users.

The difference of block sizes largely impacts the HDD access performance.

- Large block size
  - The throughput is high because data can be read continuously due to the infrequent seek times and rotation wait times.
  - The IOPs value is low because of a longer data transfer time.
- Small block size
  - The throughput is low because of the frequent seek times and rotation wait times.
  - The IOPs value is high because of a shorter data transfer time.

2.1.3 Performance with the Combination of Access Pattern and Block Size

Generally, random access with a small block size has a high IOPs value and sequential access with a large block size has a high throughput.

2.2 Performance Characteristics of HDD Storage Systems

HDD storage systems have the following two characteristics:

- Performance characteristics combining access pattern and block size
- Performance characteristics regarding HDD response time

2.2.1 Performance Characteristics Combining Access Pattern and Block Size

The performance of an HDD storage system was measured. The sequential access throughput was 294.0MB/s with a block size of 256KB (hereinafter, 256KB sequential). In the same configuration, the random access performance was 500 IOPs with a block size of 8KB (hereinafter, 8KB random).

With 500 IOPs, the throughput of 8KB random access is calculated as: 3.9MB/s (8KB × 500 IOPs)

Access Method and Block Size	IOPs	Throughput
256KB sequential		294.0MB/s
8KB random	500 IOPs	3.9MB/s (8KB × 500 IOPs)

Table 2-1 Performance Value of an HDD Storage System

In terms of throughput, it is ideal if 8KB random has the same performance as 256KB sequential, but throughput of 8KB random actually is 1/75 (3.9MB/s / 294.0MB/s) of 256KB sequential. Only 1/75 of the data can be transferred in the same time frame.

This indicates that the difference of the access pattern and block size for the HDD results in a large difference in random access performance. This is a characteristic of the HDD, which is universal regardless of the vendor, model, or HDDs that constitute the RAID.

2.2.2 Performance Characteristics Regarding HDD Response Time

There are great differences in the response time of an HDD because physical movement, such as that of the magnetic head, does not always take the shortest path. Although the shortest access time theoretically exists, access times concentrate around 20ms and gradually decrease to 200ms.

Although it is possible to calculate the average response time of an HDD, estimating the actual response time is difficult due to this large variance.

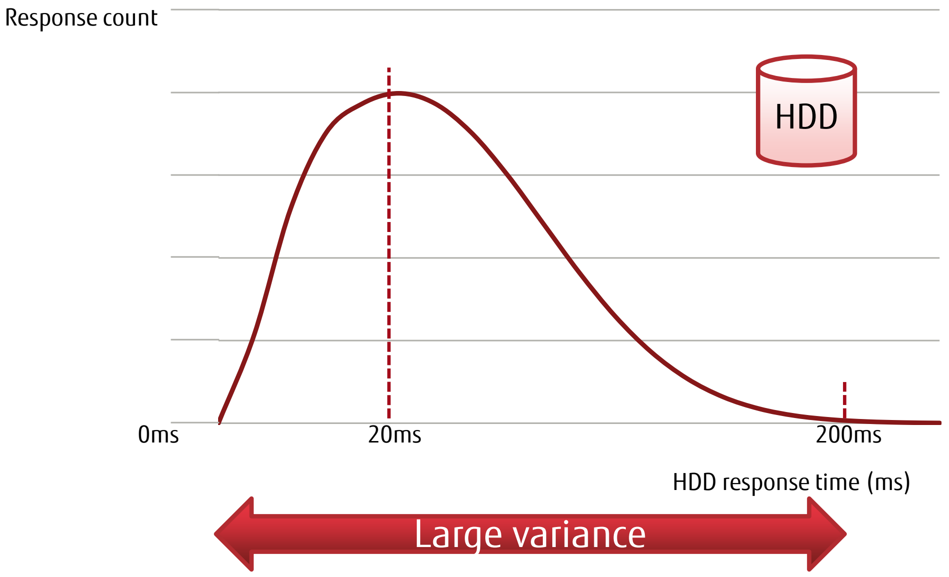


Figure 2-1 Variance in HDD Response Times

A large variance in the HDD response times impacts the time required for application processing, such as SQL process of a database. As an example, the following table shows the calculated processing times when 10,000 I/Os are issued with the most common response time at 20ms and the worst at 200ms.

The results are 200 seconds (3 minutes 20 seconds) and 2,000 seconds (33 minutes 20 seconds), which have a large difference.

HDD Response Time (ms)	Processing Time When 10,000 I/Os Are Issued (second) (HDD response time/1,000 × 10,000)
Most common case 20ms	200 seconds (3 minutes 20 seconds)
Worst case 200ms	2,000 seconds (33 minutes 20 seconds)

Table 2-2 HDD Response Time and Processing Time

It is difficult to estimate the processing time of an application that uses an HDD storage system because the access time of the highest concentration and the processing time of the worst case have too large a difference.

3 SSD Innovation of Storage Systems

SSDs fundamentally resolve the difficulty of estimating the performance of storage systems that use HDDs.

A storage system that uses SSDs has 1,000 times the performance of those that use HDDs, as well as a relatively constant performance that is not affected by the access pattern and access load of user applications. Because SSDs are memory devices, they do not move physically, and therefore their performance is not much affected by the difference of block sizes.

3.1 Performance Characteristics of Flash Storage Systems

Estimations are easy due to the following:

- Performance characteristics combining access pattern and block size
- Performance characteristics regarding SSD response time

3.1.1 Performance Characteristics Combining Access Pattern and Block Size

We measured the performance of a flash storage system with 256KB sequential and throughput of 490MB/s. In the same configuration, it represented 27,765 IOPs with 8KB random.

With 27,765 IOPs, the throughput of 8KB random access is calculated as: 216.9MB/s (8KB × 27,765 IOPs).

Access Method and Block Size	IOPs	Throughput	[Reference] HDD throughput
256KB sequential		490.0MB/s	294.0MB/s
8KB random	27,765 IOPs	216.9MB/s (8KB × 27,765 IOPs)	3.9MB/s (8KB × 500 IOPs)

Table 3-1 Performance Value of a Flash Storage System

The throughput of 8KB random is about 1/2 (216.9MB/s / 490.0MB/s) of 256KB sequential, which is less significant compared to 1/75 (3.9MB/s / 294.0MB/s) of the HDD. This indicates that with the SSD, the difference of the access pattern and block size does not result in much difference in performance compared to the HDD.

3.1.2 Performance Characteristics Regarding SSD Response Time

The SSD has a small variance in the response times. The figure below shows the distribution of the access time of random access with a small block size, which concentrates around 0.5ms.

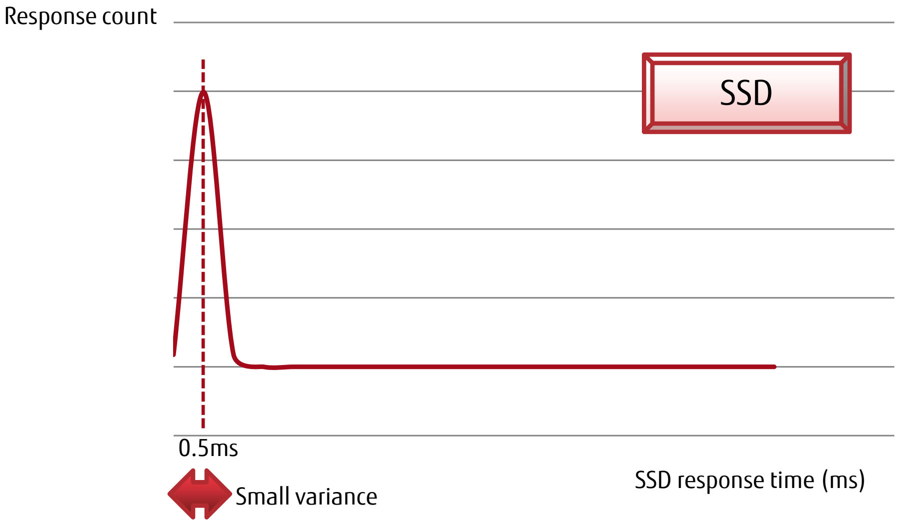


Figure 3-1 Variance in SSD Response Times

The figure below compares the variance in the response times of the SSD and HDD.

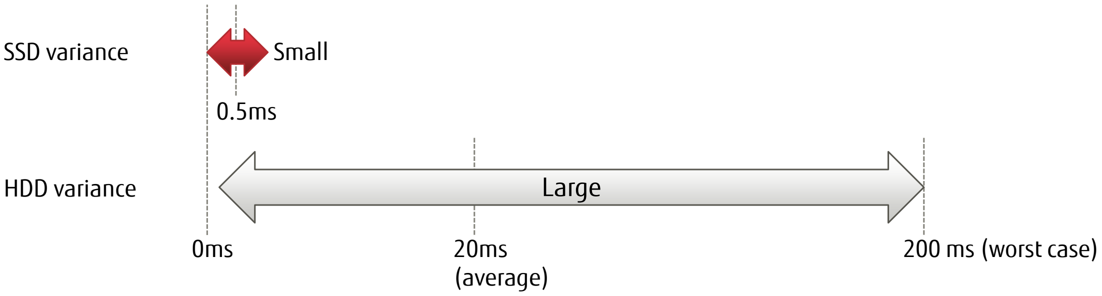


Figure 3-2 Variance in Response Times (comparison between SSD and HDD)

The figure shows that the variance in the SSD response times is small and that of the HDD is large.

As an example, the following table shows the calculated processing time when 10,000 I/Os are issued with the response time at 0.5ms. Because the variance is small and the processing time is short, the processing time concentrates around 5 seconds.

SSD Response Time (ms)	Processing Time When 10,000 I/Os Are Issued (second) (SSD response time/1,000 × 10,000)
0.5ms	5 seconds
[Reference] HDD response time 20ms to 200ms	200 seconds (3 minutes 20 seconds) to 2,000 seconds (33 minutes 20 seconds)

Table 3-2 SSD Response Time and Processing Time

The processing time can be estimated because of the small variance in the SSD response times.

3.1.3 Flash Storage Systems Providing Stable Performance

As described above, SSDs provide stable, high performance regardless of the access method, block size, and other conditions compared to HDDs. In terms of throughput, the SSD provides 1.7-fold better performance with 256KB sequential and 55.6-fold better performance with 8KB random compared to the HDD.

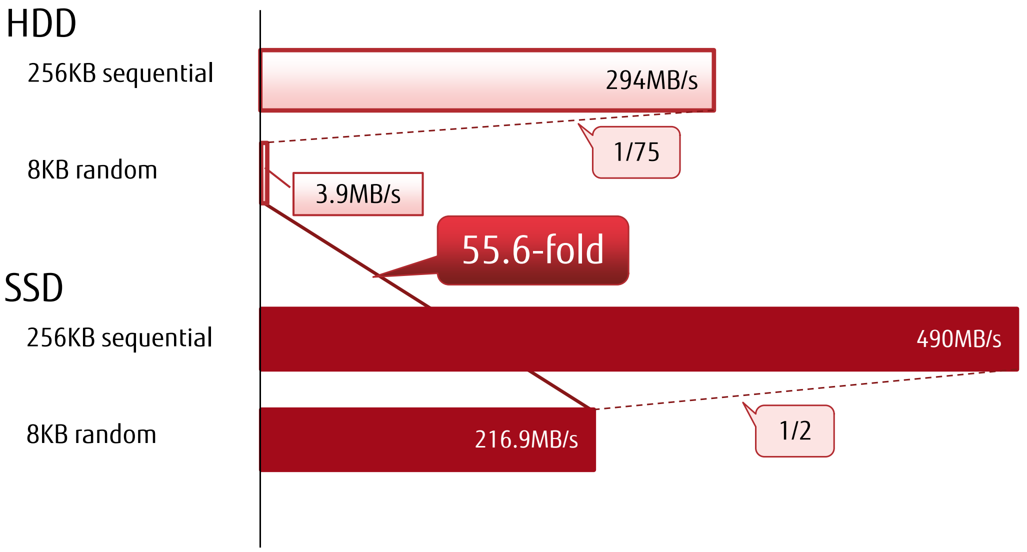


Figure 3-3 Comparison of HDD and SSD (throughput)

The SSD response time is 1/40 of the HDD (20ms) and is stable at 0.5ms.

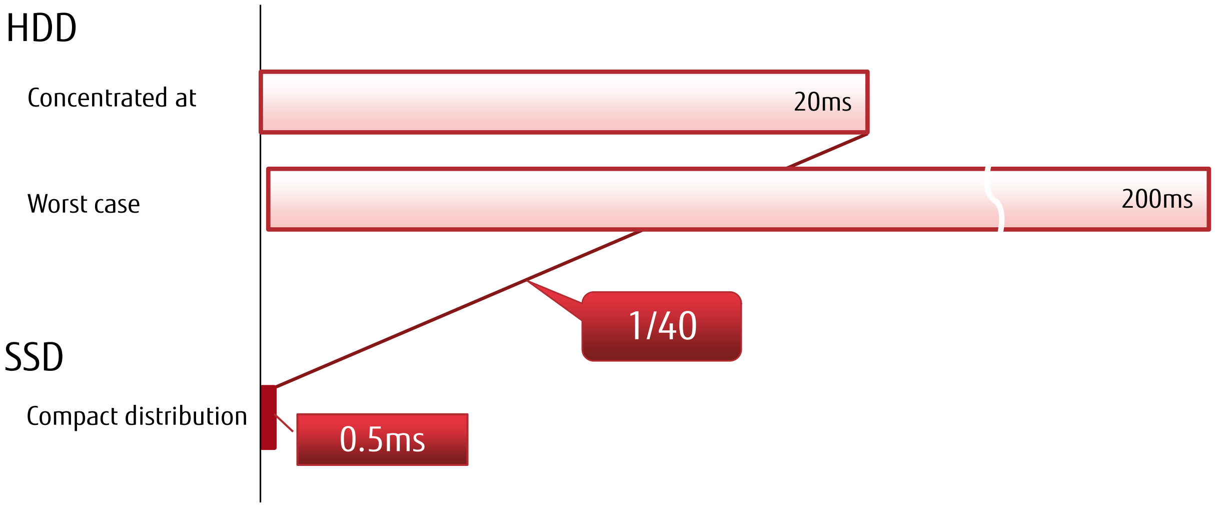


Figure 3-4 Comparison of HDD and SSD (response time)

The processing times can be easily estimated due to stable access performance. They increase in proportion to the number of times the storage device is accessed. For example, if the time to execute 100 application processes is 0.1 second, the processing time of 1 million processes is 1,000 seconds (16.7 minutes), and 10 million processes is 100,000 seconds (27.8 hours).

3.2 Systems Suitable for Flash Storage

The SSD is especially effective for a DWH that handles large amounts of data because the SSD provides a constant response time until it reaches the maximum performance.

A DWH is a database that a business enterprise has constructed by consolidating various databases chronologically for different purposes in order to conduct analyses for making decisions. A DWH is used to manipulate accumulated data, such as sorting and retrieval with different patterns, and it is used mainly for reference.

The biggest responsibility of a DWH is to process large amounts of data. Unlike OLTP, retrieving all the data from the database cache is not realistic. Of course, the database cache is tuned to achieve a high hit rate, but achieving a perfect cache hit rate is difficult for a DWH, which handles large amounts of data.

Cache misses on a database result in direct access to the storage system, which largely impacts database performance. The amount of data can be large regardless of the memory capacity of the DWH server. Therefore, many DWHs have cache misses, which affect the biggest responsibility of DWHs. This results in poor response, processing delays, and limitations on the amount of data that can be handled.

If an HDD storage is used, data retrieval after a cache miss takes a very long time because the performance of the HDD is significantly lower than that of the CPU and memory. In order to improve this situation, it has been necessary to perform complicated and meticulous database tuning as the amount of data increases.

Flash storage responds very quickly to direct storage access, which occurs frequently in a DWH. This reduces the I/O latency of the CPU, allowing the effective use of overall system resources such as CPUs and memory. Flash storage also dramatically reduces limitations on the DWH server memory capacity as well as the bothersome task of the database tuning. So DWH operations are significantly reduced.

4 Estimating SQL Processing Times with SSDs

A database with large amounts of data takes a long time for SQL process because not all data can be loaded in the memory. It sometimes takes hours. If the SQL processing time can be estimated, work efficiency can be greatly improved.

This section explains how to estimate SQL processing times.

Because the response time variance of SSDs is small, SQL processing times can be calculated from a reference SQL processing time based on the I/O count (hereinafter, I/O cost) of a reference SQL process and that of the target SQL process.

Because the SQL processing time is proportional to the I/O cost, the following equation applies:

(I/O cost of the target SQL process) × (processing time of the reference SQL process) ÷ (I/O cost of the reference SQL process) =
(processing time of the target SQL process)

4.1 Example of SQL Processing Time Estimation

The following example estimates the SQL processing time with an increased number of records stored in a table, with Microsoft SQL Server as the database software and Microsoft SQL Server Management Studio as the management software.

An SQL statement took nine seconds for a table with a record length of 11,240 bytes and with a record count of 9,452,142.

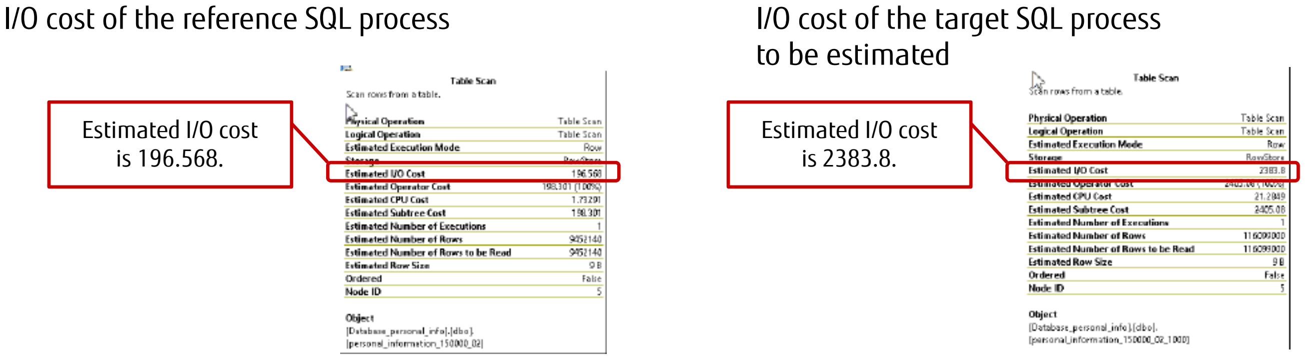
This SQL statement is used as the reference statement and its processing time as the reference SQL processing time. The following table shows the estimated processing time of the same SQL statement when the record count is increased to 116,099,340 (or about 12.28 times the original count). Data distribution is not changed in order to maintain the probability of a table extraction using the SQL statement.

	Record Count	Record Count Ratio
Reference SQL configuration	9,452,142	1
Target configuration with increased records	116,099,340	12.28

Table 4-1 Table Configuration

The following section shows how to check the I/O cost from the reference SQL file and the target SQL file by using Microsoft SQL Server Management Studio.

The I/O cost is shown in [Estimated I/O Cost], which is displayed on the details screen of [Display Estimated Execution Plan (Ctrl+L)] of the target table. The following figures show actual screenshots. The I/O cost of the reference SQL process is "196.568" and the I/O cost of the target SQL process is "2383.8" (or about 12 times greater).



I/O cost of the reference SQL process

Estimated I/O cost is 196.568.

Table Scan

Scan rows from a table.

Physical Operation	Table Scan
Logical Operation	Table Scan
Estimated Execution Mode	Row
Storage	RowStore
Estimated I/O Cost	196.568
Estimated Operator Cost	198.301 (100%)
Estimated CPU Cost	1.73291
Estimated Subtree Cost	198.301
Estimated Number of Executions	1
Estimated Number of Rows	9452140
Estimated Number of Rows to be Read	9452140
Estimated Row Size	9 B
Ordered	False
Node ID	5

Object
[Database_personal_info].[dbo].
[personal_information_150000_02]

I/O cost of the target SQL process to be estimated

Estimated I/O cost is 2383.8.

Table Scan

Scan rows from a table.

Physical Operation	Table Scan
Logical Operation	Table Scan
Estimated Execution Mode	Row
Storage	RowStore
Estimated I/O Cost	2383.8
Estimated Operator Cost	2405.08 (100%)
Estimated CPU Cost	21.2849
Estimated Subtree Cost	2405.08
Estimated Number of Executions	1
Estimated Number of Rows	116099000
Estimated Number of Rows to be Read	116099000
Estimated Row Size	9 B
Ordered	False
Node ID	5

Object
[Database_personal_info].[dbo].
[personal_information_150000_02_1000]

Figure 4-1 Checking I/O Cost

4.2 Estimation of SQL Processing Time and Result

The I/O cost of the target SQL process was 2383.8, the processing time of the reference SQL process was 9 seconds, and the I/O cost of the reference SQL process was 196.568. As a result, 2383.8 × 9 ÷ 196.568 = 109.14.

When the SQL statement was executed, the processing time was 32 seconds; less than the calculated 109.14 seconds.

5 Conclusion

Use of SSDs as storage devices in a computer system – such as a DWH, which often handles large amounts of data – reduces the processing times and provides accurate estimations of the processing times. Because of this, a wider range of information can be obtained faster and more systematically than ever before.

The ETERNUS AF series resolves performance issues of computer systems to open up the possibility of new systems.

Contact
FUJITSU Limited
Website: https://www.fujitsu.com/eternus/

■Trademarks
Microsoft®, Windows, and other Microsoft's product names are registered trademarks or trademarks of Microsoft Corporation in the United States, and other countries. Names and logos related to Red Hat are trademarks or registered trademarks of Red Hat, Inc. in the U.S. and other countries. The product names and company names in this document are registered trademarks or trademarks of their respective companies. Trademark symbols such as (R) and (TM) may be omitted from system names and product names in this document.
■Disclaimer
FUJITSU Limited is not responsible indemnity that might be caused by the contents in this documentation or any damage related to contents in this documentation.