# Outline of the SUPER-UX, Operating System for the SX-9

TOMARU Hiroko, MIYAZAKI Emiko, OHTANI Atsuhisa
SATAKE Koji, SAKAI Satoshi, KITAGAWA Toshiyuki

**Abstract**

The SUPER-UX operating system of the Supercomputer SX-9 is an enhanced version of the previous SUPER-UX system that has established reliability and achievements with the SX-6/7/8 Series.

The SUPER-UX inherits the fundamental features of previous products such as high speed, large scale system compatibility and high reliability but it is also more user-friendly and offers improved operation management of large-scale systems.

This paper summarizes the features of the SUPER-UX system and the enhanced advantages of GFS and NQSII.

**Keywords**

supercomputer, operating system, multi-node, cluster, gStorageFS, JobManipulator, SCACCT

## 1. Introduction

The progress of hardware technology and the resulting improvements in the computing performance and cost efficiency of supercomputers have expanded the field of their applications over a wide range, from major users such as governmental and university computing centers to private businesses as well as in individual laboratories.

In the field of High Performance Computing (HPC), the number of tasks to be processed is limited within a node. Consequently, the mainstream method recently adopted when it is required to increase the number of parallel tasks is the multi-node system. This system increases the number of nodes and the amount of parallel processing within a node. The number of nodes per multi-node system is therefore tending to increase.

This trend makes it no longer sufficient for the supercomputers to be capable of increasing the scale and speed of the system. It is now becoming more important for them to have the ease and flexibility of the series of operations improved, from system installations to operational management, program developments and to offer compatibilities with standardizations and open system implementations.

In the following sections, we will discuss the features of the SUPER-UX, including its large scale capability, high reliability, high speed, ease of operation, substantial operational management and its compatibility with new standards and open systems. We will also introduce details of some of the recent enhancements ( **Fig. 1** ).

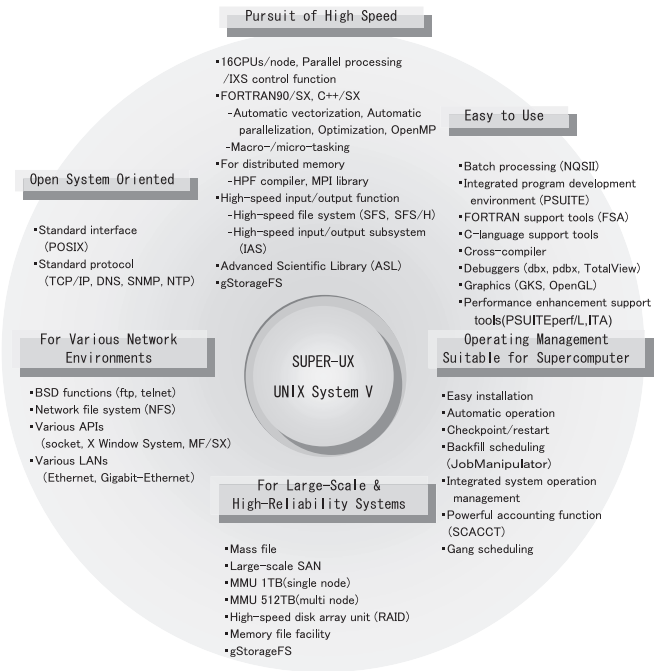



Fig. 1 Features of SUPER-UX.

## 2. Features of the SUPER-UX

The SUPER-UX is an operating system based on the UNIX

System V operating system that features functions inherited from the BSD and SVR4 2MP as well as enhancements of functions required to support supercomputers.

The SUPER-UX of the SX-9 can be run as it is on the SX-6/7/8 Series, except that its functions may be restricted depending on differences in the hardware. This feature allows the latest functions to be used on more SX Series products and also guarantees migration from the SX-6/7/8 series to the SX-9 without losing compatibility, as well as continuing to offer the ease of combination of previous models.

With the SUPER-UX, the kernel itself is highly parallelized in order to support a maximum of 16 CPUs with a single-node system or a maximum of 8,192 CPUs with a multi-node system in which a maximum of 512 nodes are clustered.

The memory in the SX-9 has been expanded from a maximum of 258G bytes to 1T bytes with a single-node system. This means that the maximum memory size of a multi 512-node system is 512T bytes.

As seen above, the SUPER-UX features flexible resource management and the high parallel processing capabilities of kernels and I/Os in order to guarantee high scalability from single-node 16 CPU systems to large-scale node systems.

The operation tools for the SX-9 are enhanced in order to avoid complications in program execution and management, even when the number of nodes is increased.

### 2.1 Large–scale/High–reliability System Compatibility

**(1) Large-scale Memory**

**1) Supporting 64M bytes Page**

Three page sizes are supported, which are the 32K bytes for general commands, the 4M bytes for compiler and system commands and the 64M bytes for large-scale user programs. This strategy improves the execution performance of programs that use large arrays and reduces the overheads in the memory management.

**2) Expansion of Virtual Space**

SUPER-UX of the SX-9 has expanded the user virtual space from about 800G bytes previously to 4T bytes ( **Fig. 2** ).

This expansion has enabled an efficient layout of a parallel program and a MPI program, the former uses an entire 1T byte main memory to process a single task, and uses the huge global memory.

**(2) Memory File Facility**

SX-MFF (SX Memory File Facility) is provided, which can enable high-speed I/O by building an ordinary file system on the large-capacity memory and can be used as the disk cache.

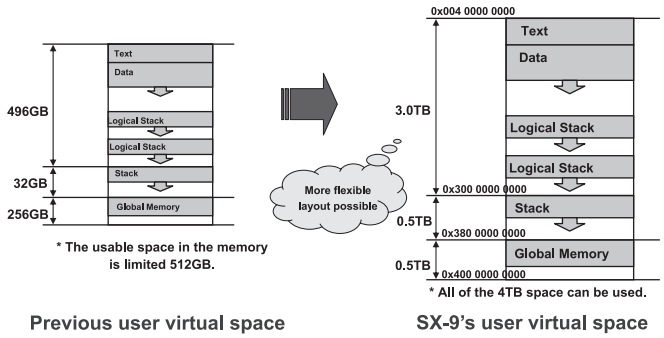



Fig. 2 4T-byte user virtual space.

### 2.2 Pursuit of High Speed

**(1) IXS Control Function**

An ultra high-speed internode connection device called IXS (Internode Crossbar Switch) is used to support the multi-node systems.

When message transfer between nodes is generated in the MPI (Message Passing Interface) or HPF (High Performance Fortran), it is usually necessary to call the OS in order to execute the transfer. However, the IXS control function supported by the SUPER-UX allows the user programs to transfer data directly to another node without intermediation of OS, so that high performance can be achieved by effectively running distributed parallel programs in MPI or HPF.

In addition, TCP/IP (Transmission Control Protocol/Internet Protocol) is implemented on IXS in order to quickly achieve file transfer using ftp (file transfer protocol) and file sharing using NFS.

**(2) High-speed File Sharing in Large-scale SAN**

High-speed file sharing on multi-platforms including the SX Series and Linux machines is made possible in the large-scale SAN (Storage Area Network) environment using fiber channels.

### 2.3 Pursuit of Ease of Use

- **IOX Software**

The IOX (Integrated Operation Station for SX) software can install the node from the web regardless of the single-node or multi-node system. This makes it possible to reduce the time taken for installation to startup.

The modified item distribution tool can be used to distribute and apply a modified item to all nodes in a multi-node

system and to refer to the application situation for improved maintainability.
SUSE Linux Enterprise Server is adopted as the IOX to support the duplication using EXPRESSCLUSTER X. The resulting redundant configuration enables continued operations of the IOX software and the applications executable on IOX such as the job scheduler.

## 3. High–speed I/O System

As a general rule the HPC systems do not simply use the SX series but incorporate them into a large variety of machine environments including front-end servers and other computing nodes such as scalar machines. This trend sometimes creates a problem in the time taken and in the complexity of migration of large-capacity data between different machines, for example between computing nodes and front-end servers.

On the other hand, the I/O system is also required to offer the higher speed and higher efficiency that matches the CPU's processing capability and the memory capacity.

In this section, we will discuss NEC's approach to the I/O component of HPC systems.

### 3.1 Purpose of GFS

In order to solve the problems related to the handling of large-capacity data as mentioned above, NEC's gStorageFS (hereinafter abbreviated to GFS) provides a high-speed file sharing function [1] assuming the use of fiber channel-based SAN. This enables seamless, high-speed data access without migrating large-capacity data even in a heterogeneous environment as shown in **Fig. 3** .

For example, when the end user logs in a front-end server, places the input data required for a job in GFS and inputs the job, a computing node of the SX Series loads the input data from GFS and writes the computing results in GFS. When the job has been executed, the end user can refer the computing results from the GFS through the front-end server, so that the end-user may proceed to the next processing task such as visualization without migrating data. Since the GFS access is performed via the fiber channel-based SAN, the access speed is much higher than access via the network.

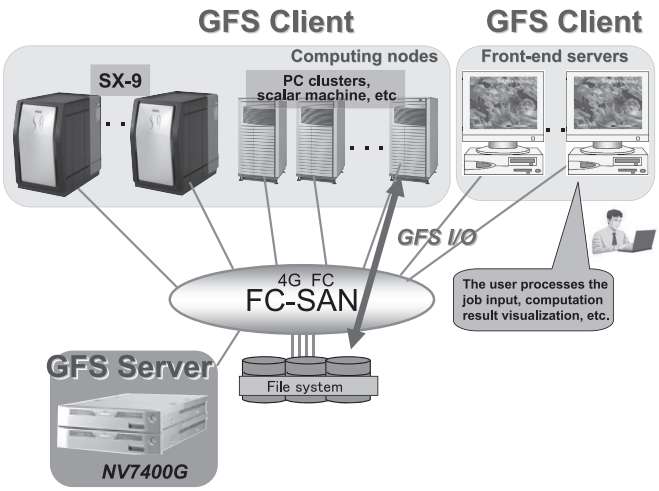


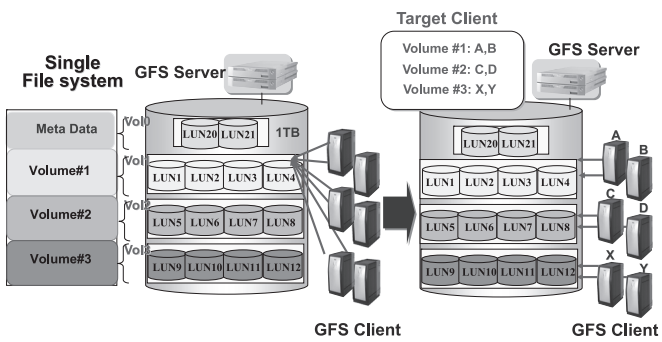

Fig. 3 Concept of file sharing configuration.

### 3.2 Efficiency Improvement Attempts for Large–scale File System and Parallel I/O

The GFS can build and operate a large-scale file system up to 128T bytes by combining the concatenation of the striping and striping disk into a single file system.

GFS allows direct disk access from GFS clients. In order to manifest performance efficiently in a large-scale file system, it is required that the accesses from all of the nodes of the GFS clients are distributed equally to all of the disks of the file system. The distributed file layout function, by defining combination of the "GFS clients" and the "volumes in which the GFS clients create files in priority," is in order to avoid concentration of I/O to specific volumes and to improve the overall system throughput.

**Fig. 4** shows the cases in which multiple nodes perform I/O accesses simultaneously to a single file system when the distributed file placement function is used or not used. Case (a) shows the concentration of accesses in a single volume, while




(a) Without the distributed layout function (b) With the distributed layout function

Fig. 4 Distributed file placement of GFS.

case (b) shows the distribution of accesses to multiple volumes without causing performance degradation.

## 4. Advanced Job Execution Environment

The SX Series allows small- to large-scale configurations to achieve high availability of high-performance system resources and provides an advanced job execution environment that can execute applications easily at high speeds and with high reliability. The job execution environment features an advanced resource scheduling function that assigns the ultra-high-speed computing performance of the SX Series application execution to the limit, as well as an accounting system enhanced to support multi-node compatibility.

### 4.1 Batch Scheduler (JobManipulator)

JobManipulator is in charge of the scheduling of the NQSII (Network Queuing System II). JobManipulator implements a job management system that offers high system availability and resource optimization in conjunction with NQSII.

**(1) Backfill Scheduling**

JobManipulator provides backfill scheduling based on the declarations of the required resource amount (the number of CPU, memory volume, etc.) and the required computing time (elapsed time), which is made by the user at the time of job input. Backfill scheduling implements the resource occupation that guarantees high system availability and high-speed job execution. Backfill scheduling refers to the scheduling that optimizes the projected assignments of system resources to jobs for limited periods. It improves the availability of computing nodes and also optimizes the resource usage of the entire system resources by placing jobs efficiently in a multidimensional space (resource space). Thus representing the system resource management using the axis in the real-time direction, the axis representing the computing nodes and the axis representing resources (JobManipulator uses two axes, which represent the number of CPUs and the memory volume, for this purpose) ( **Fig. 5** ).

**(2) Various Scheduling Functions**

JobManipulator offers job management solutions that can deal flexibly with a large variety of user needs.

- Fair-share scheduling achieves equal distribution of resources according to the resource usage share set on a per-use or per-group basis and the past resource usage history.
- Scheduling priority controls the job execution order dynamically according to more than 10 items and to the priority that is calculated dynamically as the sum of weighting of the items.
- An emergent job execution function is also provided for jobs to be executed according to highest priority. With this function, the mode for emergent execution by interrupting execution of low-priority jobs and that for starting execution as soon as the executing jobs are finished can be selected.
- Advance reservation can reserve the resource space in advance in order to secure the resources required for a specific job and guarantee its start time.
- The run limit/assign limit function can set the limitation of the number of simultaneously executed jobs (run limit) and the limitation of the number of jobs assigned to the resource space (assign limit) on a per-system or per-queue basis. It also provides a complex queue function that sets the run limit and assign limit to a set of multiple queues.

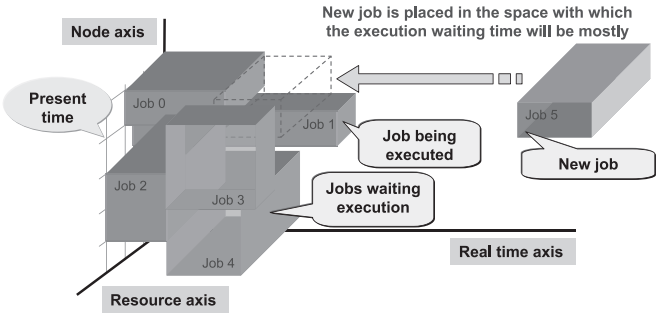



Fig. 5 Concept of back filling.

### 4.2 Multi–node Compatible Accounting System: SCACCT

Previous accounting summary had to handle a large volume of accounting records in order to compile daily and monthly accounting reports and the counting required consumption of the CPUs at each of the SX nodes. In contrast, SCACCT assigns the high-performance CPUs in the computing nodes to the applications so that the daily and monthly accounting reports can be compiled without using the resources of each SX node by off-loading the counting function from the computing node to the SCACCT server. This function is enhanced further in order to facilitate accounting operations even in a large-scale multi-node configuration.

**(1) Multi-node Compatible Accounting**

The SCACCT server implements the accounting function of

the entire multi-node system based on the centralized management of accounting information. For example, the server can easily display the whole of the accounting information for batch requests executed across multiple computing nodes.

**(2) Multi-node Compatible Budget Management**

The budget management that manages each user, group or project so that its usage does not exceed the preset amount (= budget) is given the multi-node compatibility. When it is interlocked with NQSII, for example, the job input to the entire multi-node system can be prohibited if the budget is exceeded during execution of a job on any computing node.

## 5. Conclusion

In the above, we introduced SUPER-UX, the operating system for the SX-9 supercomputer. In order to let the hardware exhibit its maximum performance, it is required that the software technologies such as the operating systems advance in step with the progress of the hardware technologies. It is expected that the future supercomputers will expand the fields of applications and the requirements of their operating systems will be more challenging than ever. It is our intension to make every effort to further advance the SUPER-UX by comprehensive predictions of user needs as well as by closely monitoring both market and technology trends.

*UNIX is a registered trademark of The Open Group.

*Ethernet is a trademark of XEROX Corporation, USA.

*Linux is a registered trademark or trademark of Linus Torvalds in the United States and other countries.

*NFS is a trademark of Sun Microsystems, USA.

*SUSE is a registered trademark of Novell, Inc.

### Reference

1) Ohtani, A. et al., “A File Sharing Method for Storage Area Network and Its Performance Verification”, NEC Res. & Develop., Vol.44, No. 1, pp. 85-90, Jan. 2003.

### Authors' Profiles

**TOMARU Hiroko**
Manager,
1st Computers Software Division,
Computers Software Operations Unit,
NEC Corporation

**MIYAZAKI Emiko**
Assistant Manager,
1st Computers Software Division,
Computers Software Operations Unit,
NEC Corporation

**OHTANI Atsuhisa**
Assistant Manager,
1st Computers Software Division,
Computers Software Operations Unit,
NEC Corporation
Member of IEICE (Institute of Electronics, Information and Communication Engineers) and RSSJ (Remote Sensing Society of Japan).

**SATAKE Koji**
Assistant Manager,
Software Development Division,
NEC Software Tohoku, Ltd.

**SAKAI Satoshi**
Technical Manager,
Server Software Division,
Platform Operations Unit,
NEC System Technologies, Ltd.

**KITAGAWA Toshiyuki**
Technical Expert,
Server Software Division,
Platform Operations Unit,
NEC System Technologies, Ltd.