# Rapidly Growing Linux OS: Features and Reliability

● Norio Kurobane

*(Manuscript received May 20, 2005)*

**Linux has been making rapid strides through mailing lists of volunteers working in the Linux communities. These volunteers help develop source code, provide usage results, report information about problems in the communities, and quickly provide required bug fixes. The latest kernel version (version 2.6) reflects many improvements that have been made to Linux, primarily in memory management and process scheduler functions. Consequently, Linux has become a much more advanced OS. This paper describes the following four features being developed by Fujitsu in conjunction with the Linux communities for mission-critical systems: 1) diskdump for reliably collecting dumps at kernel crashes and hang-ups; 2) an enhanced machine check architecture (MCA) for minimizing the effects of hardware failures and recovering from failures with a machine check facility; 3) udev: a persistent device naming feature that ensures a device name before and after maintenance or expansion; and 4) hot-plug for performing hot system maintenance of the CPU, memory, and I/O bus and expanding the system space.**

## 1. Introduction

Linux has been making rapid strides through mailing lists of several tens of thousands of volunteers working in the Linux communities. These volunteers help develop source code, provide usage results, report information about problems in the communities, and quickly provide required bug fixes. Particularly, in recent years, features required for enterprise applications have rapidly been jointly developed with corporate engineers, and many of these features have been incorporated into Linux. Accordingly, the latest kernel version (version 2.6) reflects functional improvements that have been made to Linux, primarily in the scheduler, to efficiently operate a large-scale symmetric multiple processor (SMP) and handle massive I/O devices. Furthermore, exclusive operation regions around the CPU have been drastically reduced. Linux has provided a variety of features to build large-scale systems that previously could only be achieved on mainframes or large UNIX servers. Fujitsu has been developing the features required to apply Linux to mission-critical systems in conjunction with the Linux communities and enhanced the features in the latest kernel version.

This paper describes the four features being developed by Fujitsu in conjunction with the Linux communities for mission-critical systems. The four features are diskdump (a crash dump feature), an enhanced machine check architecture (MCA: for enhanced hardware reliability, availability, and serviceability [RAS]), udev (a hardware naming feature), and hot-plug (for hot system maintenance).

## 2. diskdump

Conventionally, volunteers developed Linux by adding features and providing bug fixes. When problems occurred, the volunteers could easily take the necessary action because they understood their own work and could generally identify the part and action of the program. In addition, they could perform reproduction tests and narrow down the failure location. Therefore, there were sufficient Linux tools for investigating a failure concurrently with reproduction tests.

Recently, Linux is increasingly being used for corporate mission-critical systems and server OSs. These systems typically start several applications and concurrently process requests from a great number of clients. If a problem occurs in such a system, the volunteer developers cannot identify what was processed and can rarely reproduce the problem. It is more likely that the cause of a failure will not be determined from just the console information that is output when a failure occurs. In Linux server OS operation, the system often outputs no information when it hangs up. However, the crash dump feature is effective for collecting information about the CPU registers and memory. The information collected with the crash dump feature allows developers to reference kernel control table data, identify memory inconsistencies, and determine the cause of failures. Therefore, it is very important to install the Linux crash dump feature in corporate systems.

In the core Linux communities, the effectiveness of the crash dump feature was hardly recognized because many program developers personally used Linux. Moreover, when a failure occurred in the kernel, it seemed unlikely that accurate dump information could be collected using the kernel's features. Then, Fujitsu developed diskdump in conjunction with Red Hat, Inc., which is one of the main Linux distributors. This feature allows developers to reliably collect dump information even when a kernel error (panic or oops) or hang-up occurs. **Figure 1** shows the concept of diskdump, the main features of which are as follows:

1) diskdump minimizes the use of the kernel features at failures. For example, it allocates the area used for dump information beforehand and inhibits asynchronous events such as interrupts by suppressing other operations.
2) diskdump enables collection of dump information even during a temporary hardware error by resetting the device whose dump information is to be output.

The diskdump feature is being generalized through the promoted distribution of Red Hat Enterprise Linux AS (v.4 for Itanium), etc. Fujitsu has been working to standardize the Linux kernel features in the Linux communities. Furthermore, it has started up the lkdump community[1] so that diskdump will be widely accepted for the dump feature of mission-critical systems.

## 3. Enhanced MCA

In mainframes and UNIX servers used for mission-critical tasks, the hardware and OS work together to localize hardware failures that occur in the processor or memory and recover from the failures. This linkage helps prevent these failures from spreading over an entire system.

This section describes a feature for localizing the effects of a hardware failure and recovering from the failure. This feature has been added to Linux kernel version 2.6.

The mission-critical IA server PRIMEQUEST uses the Intel Itanium 2 processor. To improve

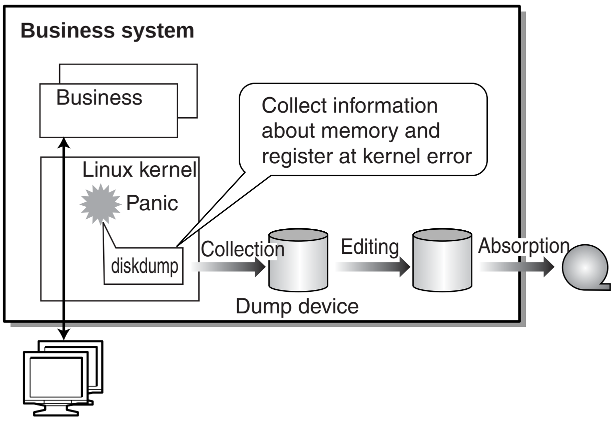

Figure 1
Concept of diskdump.

system availability and reliability, Itanium 2 has a more expanded self-diagnostic and recovery facility for hardware failures that occur in the CPU, memory, chipset, and bus than conventional IA servers. When a hardware failure occurs, this facility first tries to recover the hardware and firmware layers. If recovery succeeds, the software processing that was interrupted due to the failure is resumed. In this case, the facility notifies the OS of the corrected machine check interrupt (CMCI) or corrected platform error interrupt (CPEI), and the OS records the notified error information as log data. If recovery fails, the facility notifies the OS of the MCA and asks the OS to perform recovery processing (**Figure 2**).

The OS analyzes the error information (System Abstraction Layer [SAL] error record) that the firmware created to perform recovery processing for each error type.

Fujitsu continued discussions in the Linux communities while utilizing the know-how it accumulated when developing OSs for mainframes and showing the need to enhance the MCA and explaining how to install it. Consequently, Fujitsu succeeded in incorporating an enhanced MCA feature into Linux kernel version 2.6. This feature enables recovery processing from error correcting code (ECC) multi-bit errors that occur in the memory.

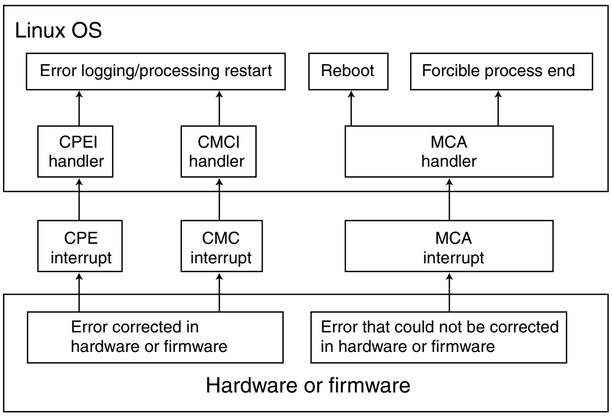
Linux OS
Error logging/processing restart
Reboot
Forcible process end
CPEI handler
CMCI handler
MCA handler
CPE interrupt
CMC interrupt
MCA interrupt
Error corrected in hardware or firmware
Error that could not be corrected in hardware or firmware
Hardware or firmware

Figure 2
Concept of MCA.

If an ECC multi-bit error occurs while a user process is reading memory data, the feature does not reboot the server. Instead, it forcibly ends the user process (with sigkill) and removes the memory page in which the error occurred from the areas to be newly allocated. As a result, the Linux kernel has the same excellent RAS as a mainframe.

If an ECC multi-bit error occurs in kernel-mode operation, the system is rebooted because the data being used by the kernel cannot be guaranteed.

A parity error on the PCI bus is also judged as recoverable among the MCA events that may occur in the Itanium 2 processor. Such an error must be recovered in consideration of the I/O request affected by the error. Therefore, the OS-MCA handler and device driver must be linked. Presently, an I/O access interface is being studied to notify the device driver of a PCI bus parity error that is detected in the OS-MCA handler. The Linux communities have been working to incorporate the I/O access interface feature in the standard kernel in conjunction with vendors who are particularly interested in it, so this feature will soon be incorporated.

## 4. udev

A UNIX OS assigns a pair of integers called a major number and a minor number to each I/O device connected to the system and identifies individual I/O devices using these pairs. This method is manageable for OS programs, but unmanageable for system administrators. Therefore, the OS relates a special file called a device node to the integer pairs that the OS has assigned to the I/O devices. The system administrator can then manage an I/O device by using the device node instead of the pair of major and minor numbers. Because Linux has a UNIX-like kernel structure, it uses a similar I/O device management method as UNIX. The OS uses a pair of static major and minor numbers to manage an I/O device, while the system administrator uses a device node that corresponds to a pair to manage an I/O device.

This method was effective in servers having a relatively smaller configuration with less I/O devices connected to the system. Recently, however, as Linux is being installed in large-scale servers with enormous numbers of I/O devices connected, some problems have occurred. For example, too many devices lead to a lack of major and minor numbers. Also, when a device is disconnected, the numbers assigned to subsequent devices deviate from the original ones, causing a collapse of the correspondence between device nodes and devices. It has therefore become difficult to respond to new environments by extending the existing method.

In the latest Linux kernel version (version 2.6), to overcome the lack of major and minor numbers, each field size has been expanded so that sufficient numbers can be assigned to I/O devices. Also, to solve the problem of a collapse of correspondence between device nodes and I/O devices, the udev feature, which manages the relationship between pairs of major and minor numbers and device nodes, has been introduced.

The udev feature is a program for creating device nodes that correspond to I/O devices according to the rule set that is defined by the system administrator. Defining an appropriate rule set helps to relate a fixed device node to an I/O device. However, a method of uniquely identifying an I/O device is still required because fixed major and minor numbers cannot be assigned to an I/O device. To give a simple example, a 48-bit unique identification code called the media access control (MAC) address is assigned to a LAN card. The MAC address can be used as an identifier to uniquely identify a LAN card. Similarly, a SCSI disk or fiber channel (FC) disk has an assigned unique identification code called the vital product data (VPD) that can be used to uniquely identify a SCSI or FC disk.

To improve reliability and throughput, multiple independent I/O paths can be set for the same disk. This setting is called multipath control. In this case, although each independent device node must be related to an I/O path, the disk identifier VPD cannot identify the I/O paths because they are connected to the same disk. This problem can be solved by using the I/O bus configuration to uniquely identify the I/O paths.

For PCI, the bus configuration can uniquely be identified with a group of four numbers: the segment number, bus number, device number, and function number. Also, for a multipath configuration, a different group of numbers is assigned to each I/O path, and this group can be used as an identifier for an I/O path.

## 5. hot-plug

In a mission-critical server that must provide high-reliability operation, hardware components are treated as modules, enabling module replacement and expansion without stopping the entire system. The hot-plug feature allows engineers to replace and expand the hardware modules while the system is on. Making these modules redundant means that system operation is unaffected when a single failure occurs in a module. In fact, when the hardware self-diagnostic feature detects a symptom of a module failure, the hot-plug feature allows engineers to preventively replace the module before it stops; this operation is called hot system maintenance of hardware.

Another advantage of treating hardware components as modules is that the CPU, memory, and I/O modules required for system operation can be grouped and each group can be used as an independent system. This mode of operation is called hardware partitioning. Recently, to reduce the total cost of ownership (TCO), servers and storages are being virtualized so their hardware resources can be collectively pooled and allocated for capacity-on-demand operation. Hardware partitioning technology is an infrastructure feature of server virtualization technology.

To increase or decrease hardware resources during system operation, new features generically called hot-plug features have been added to the Linux OS. Three different types of hot-plug features are provided for three different types of resources: CPU hot-plug, memory hot-plug, and I/O hot-plug. In some cases, different hardware resources are installed in a module for which hot system replacement or expansion is possible; for example, a module may contain a CPU and memory. A higher feature called node hot-plug is used to group the resources for hot-plug.

Currently, the Linux communities are energetically developing the hot-plug feature in conjunction with other vendors, and Fujitsu is a major member in many of the Linux communities. Linux kernel version 2.6 already supports some of the hot-plug features, which will officially become available in the next kernel version.

**Norio Kurobane** received the B.E. degree in Electrical Engineering from Tokyo University, Tokyo, Japan in 1977. He joined Fujitsu Ltd., Tokyo, Japan in 1977, where he has been developing and supporting operating systems (OSs) for mainframes, supercomputers, and fault-tolerant communications processors, for example, Linux OSs for mission critical areas. He is a member of the Information Processing Society of Japan (IPSJ).

## 6. Conclusion

The improvement of Linux's features has been accelerated thanks to the participation of server vendor engineers in addition to the conventional development by the several tens of thousands of volunteers. This paper described the enhanced features that have been supported in the latest Linux kernel version (version 2.6). Fujitsu has assumed a leading role in the development of features in conjunction with the Linux communities. Fujitsu will continue in this leading role and vigorously work with new functional improvements to expand the use of Linux in large-scale, mission-critical applications.

This research has been partially funded by the Ministry of Economy, Trade and Industry (METI) and the New Energy and Industrial Technology Development Organization (NEDO).

## Reference

1) Website of the diskdump community (lkdump). *http://sourceforge.net/projects/lkdump/*